



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

NÁVRH VYBRANÉ ČÁSTI STANDARDU IEEE 802.1Q

DESIGN OF SELECTED IEEE 802.1Q STANDARD PARTS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Filip Kliment

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Lukáš Fajcik, Ph.D.

BRNO 2018

Diplomová práce

magisterský navazující studijní obor **Mikroelektronika**
ÚSTAV MIKROELEKTRONIKY

Student: Bc Filip Kliment

ID: 164875

Ročník: 2

Akademický rok: 2017/18

NÁZEV TÉMATU:

NÁVRH VYBRANÉ ČÁSTI STANDARDU IEEE 802.1Q

POKYNY PRO VYPRACOVÁNÍ:

Cílem diplomové je vytvořit návrh funkčního IP jádra standardu IEEE 802.1QBU v jazyku VHDL. Cílem dále bude úprava MAC vrstvy pro podporu preempce ethernetových rámců popsané ve standardu IEEE 802.3BR. Celé jádro bude používat rozhraní Avalon Streaming (rozhraní podporované Intel FPGA). IP jádro bude testované simulací nástrojem Modelsim s použitím vlastních verifikačních BFM

DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce

Termín zadání: 14.6.2018

Termín odevzdání: 20.8.2018

Vedoucí práce: doc. Ing. Lukáš Fucík, Ph.D.

Konzultant:

doc. Ing. Lukáš Fucík, Ph.D.
předseda oborové rady

UPOZORNĚNÍ:

Autor semestrální práce nesmí při vytváření semestrální práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRACT

This thesis deals with network substandards from the TSN group (IEEE 802.1Q), which deal with prioritization of network traffic in TSN networks. These sub-standards include 802.1QBV and 802.1QBU, which have been described in more detail and compared in terms of network permeability and latency. Substandard 802.1QBU was chosen for the design implementation in FPGA. The design was described in VHDL. The developed design was verified by simulations, using self-tests. The work includes synthesis and time analysis.

KEYWORDS

Ethernet, TSN, IEEE, OSI, MAC, Frame preemption, Data Link Layer, FPGA, VHDL

ABSTRAKT

Táto práca sa zaoberá sieťovými podštandardmi z kategórie TSN (IEEE 802.1Q), ktoré riešia prioritizáciu sieťovej prevádzky v TSN sieťach. Medzi tieto podštandardy patria 802.1QBV a 802.1QBU, ktoré boli detailnejšie popísané a porovnané z hľadiska priepustnosti a latencie. Podštandard 802.1QBU bol zvolený pre návrh implementácie do obvodu FPGA. Dizajn bol popísaný v jazyku VHDL. Navrhnutý dizajn bol overený simuláciami, s využitím samočinných testov. V práci sú zahrnuté aj výsledky syntézy a časovej analýzy.

KLÍČOVÁ SLOVA

Ethernet, TSN, IEEE, OSI, MAC, Frame preemption, Data Link Layer, FPGA, VHDL

KLIMENT, F. NÁVRH VYBRANÉ ČÁSTI STANDARDU IEEE 802.1Q . Brno:
Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií,
2018. 64 s. Vedoucí diplomové práce doc. Ing. Lukáš Fojcik, Ph.D..

Prehlásenie

Prehlasujem, že svoju diplomovú prácu na Návrh vybrané časti standardu IEEE 802.1Q som vypracoval samostatne pod vedením vedúcej / ho diplomovej (bakalárskej) práce a s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce .

Ako autor uvedenej diplomovej (bakalárskej) práce ďalej prehlasujem, že v súvislosti s vytvorením tejto diplomovej (bakalárskej) práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Zb., vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákonníka č. 40/2009 Zb.

V Brně dne: **20. srpna 2018**

.....
podpis autora

Pod'akovanie

Ďakujem vedúcemu diplomovej práce doc. Ing. Lukáš Fucík, Ph.D. a konzultantovi diplomovej práce Ing. Petr Grillinger, Ph.D. za účinnú metodickú, pedagogickú a odbornú pomoc a ďalšie cenné rady pri spracovaní mojej diplomovej práce. Ďalej chcem poďakovať spoločnosti TTech za poskytnutie prostriedkov k vypracovaniu tejto diplomovej práce.

V Brně dne: **20. srpna 2018**

.....
podpis autora

Experimentální část této diplomové práce byla realizována na
výzkumné infrastruktuře vybudované v rámci projektu
CZ.1.05/2.1.00/03.0072

**Centrum senzorických, informačních a komunikačních systémů
(SIX)**

operačního programu Výzkum a vývoj pro inovace.

OBSAH

Obsah	6
Zoznam symbolov a zkratiek	8
Zoznam obrázkov	9
Zoznam tabuliek	10
Úvod	11
1 Sieťová komunikácia	12
1.1 Model OSI (Open systems interconnection model).....	12
1.1.1 Fyzická vrstva	12
1.1.2 Linková vrstva	12
1.1.3 Sieťová vrstva	13
1.1.4 Transportná vrstva	13
1.1.5 Relačná vrstva.....	14
1.1.6 Prezentačná vrstva	14
1.1.7 Aplikačná vrstva	14
1.2 802.1	14
1.2.1 802.1Q.....	14
1.3 802.3	15
1.4 Efektívnosť prenosu (dátové komunikácie a vytváranie sietí)	15
2 Siete TSN (time-sensitive networking)	18
2.1 Časová synchronizácia.....	18
2.2 Sieťová konfigurácia.....	19
2.3 802.1QBV	20
2.3.1 Vytváranie chráneného okna, používaním operácií na bránach	21
2.3.2 Presný časový protokol.....	22
2.3.3 Plánovaná prevádzka a koncové stanice	22
2.3.4 Prepojenie vo vnútri 802.1QBV	22
2.4 802.1QBU	23
2.4.1 Preempcia použitá v izolácii	24
2.4.2 Plánovanie a preempcia použitá v kombinácii bez HOLD/RELEASE	

2.4.3	Plánovanie a preempcia použitá v kombinácii s HOLD/RELEASE ..	25
2.4.4	Prideľovanie šírky pásma a expresná prevádzka siete.....	25
2.4.5	Prepojenie v 802.1QBU	25
3	802.3BR	27
3.1	Formát rámcu podľa 802.3BR	27
3.2	Kódovanie SMD a čísla fragmentu.....	28
4	Teoretický výpočet a porovnanie priepustnosti 802.QBU s inými štandardmi	30
4.1	Porovnanie priemernej sieťovej prevádzky cez 802.1QBU a cez štandardný ethernet	30
4.2	Porovnanie uvažovanej sieťovej prevádzky cez ethernet, 802.1QBU, 802.1QBV a ich kombinácii	31
4.3	Porovnanie latencie	35
5	Implementácia podštandardu 802.1QBU	37
5.1	RX MAC.....	39
5.2	TX MAC	41
5.3	CRC_REMOVER.....	46
5.4	Avalon Streaming	46
5.5	Syntéza.....	47
6	Simulácie	48
6.1	Testovacie prostredie tb_top_rx.....	48
6.1.1	Test procedúra tp_preamble.....	49
6.1.2	Test procedúra tp_express	50
6.1.3	Test procedúra tp_no_preemption	51
6.1.4	Test procedúra tp_preemption	53
6.2	Testovacie prostredie tb_top_final.....	54
6.2.1	Test procedúra tp_padding.....	56
6.2.2	Test procedúra tp_back_to_back	56
6.2.3	Test procedúra tp_long_pckt	58
6.2.4	Test procedúra tp_corner_cases	58
6.3	Výsledky test procedúr	59
7	Záver	61
	Literatúra	63

ZOZNAM SYMBOLOV A ZKRATIEK

- ALUT** Adaptívna Look Up tabuľka (Adaptive Look-Up Tables)
- CD** nepretržité doručovanie (Continuous delivery)
- CRC** Cylická redundančná kontrola (Cyclic redundancy check)
- CSMA** Naslúchaniu nosný viacnásobný (Carrier Sense Multiple Access)
- eMAC** expresný MAC
- FCS** kontrolná sekvencia rámcu (Frame check sequence)
- FIFO** prvý dnu, prvý von (First in First out)
- FPGA** Programovateľné hradlové pole (Field-programmable gate array)
- GMII** Médiovo nezávislé rozhranie pre gigabitové médium (Gigabit media-independent Interface)
- IP** Duševné vlastníctvo (Intellectual Property)
- IEEE** Inštitút pre elektrotechnické a elektronické inžinierstvo (Institute of Electrical and Electronics Engineers)
- LAN** Lokálna sieť (Local Area Network)
- MAC** Kontrola prístupu k médiu (Media Access Control)
- MAN** Metropolitná sieť (Metropolitan Area Network)
- MFCS** kontrolná sekvencia pre Mrámec (Mframe check sequence)
- MII** Médiovo nezávislé rozhranie (Media-independent Interface)
- OSI** Otvorené prepojenie systémov (Open Systems Interconnection)
- pMAC** preempčný MAC
- PTP** Precízny časový protokol (Precision time protocol)
- SFD** Počiatočný oddeľovač rámcov (Start Frame Delimiter)
- SMD** Počiatočný oddeľovač mrámcov (Start mFrame Delimiter)
- SMD-Cx** Počiatočný oddeľovač mrámcov pokračujúci fragment(Start mFrame Delimiter continues fragment)
- SMD-Ix** Počiatočný oddeľovač mrámcov iníciaľny fragment(Start Mframe Delimiter Initial Fragment)
- TSN** Časovo citlivé siete (Time sensitive networking)
- VHDL** VHSIC jazyk pre popis hardvéru (VHSIC Hardware Description Language)
- VLAN** Virtuálna LAN (Virtual LAN)

ZOZNAM OBRÁZKOV

Obr. 1.1: Graf závislosti priepustnosti na veľkosti rámcu. [12]	16
Obr. 2.1: Ochranné pásmo A) pevný začiatok ochranného pásma B) premenný začiatok ochranného pásma.....	21
Obr. 2.2: Príklad operácií prebiehajúcich na bránach.....	22
Obr. 2.3: Schematické zobrazenie zapojenia pre podštandard QBV.....	23
Obr. 2.4: Príklad preempcie rámcov.....	24
Obr. 2.5: Bloková schéma pre 802.1QBU.....	26
Obr. 3.1: Porovnanie formátov klasického MAC rámca, Expresného rámca a Preemptovateľného rámca.....	27
Obr. 3.2: Formáty fragmentov preemptovateľného rámcu.....	28
Obr. 4.1: Závislosť priemernej priepustnosti na dĺžke rámca, pre štandardný ethernet a varianty podštandardu s rôznou minimálnou dĺžkou fragmentu.....	31
Obr. 4.2: Porovnanie závislostí priepustností pre štandardný ethernet QBV, QBU s minimálnou dĺžkou fragmentu 64 bajtov a kombinácie QBV s QBU s minimálnou dĺžkou fragment 64 bajtov.....	33
Obr. 4.3: Porovnanie závislostí priepustností pre štandardný ethernet QBV, QBU s minimálnou dĺžkou fragmentu 128 bajtov a kombinácie QBV s QBU s minimálnou dĺžkou fragment 128 bajtov.....	34
Obr. 4.4: Porovnanie závislostí priepustností pre štandardný ethernet QBV, QBU s minimálnou dĺžkou fragmentu 256 bajtov a kombinácie QBV s QBU s minimálnou dĺžkou fragment 256 bajtov.....	35
Obr. 4.5: Porovnanie veľkosti latencie pre expresné rámce.....	36
Obr. 5.1: Top modul implementovaného dizajnu, s prepojením vnútorných komponentov.....	38
Obr. 5.2: Diagram stavového automatu v RX MAC.....	41
Obr. 5.3: Diagram stavového automatu v TX MAC.....	45
Obr. 5.4: Príklad fungovania komunikačného rozhrania Avalon ST	47
Obr. 6.1: Bloková schéma testovacieho prostredia tb_top_rx.....	48
Obr. 6.2: Bloková schéma testovacieho prostredia tb_top_final.....	55
Obr. 6.3: Textový výstup z Code Coverage analýzy v programe Modelsim.....	60

ZOZNAM TABULIEK

Tabuľka 3-1: Hodnoty pre SMD polia a polia s číslom fragmentu. [17].....	29
Tabuľka 5-1: Tabuľka prechodov medzi stavmi pre automat v TX MAC.	40
Tabuľka 5-2: Tabuľka výstupov pre stavový automat v RX MAC.	41
Tabuľka 5-3: Tabuľka prechodov medzi stavmi pre automat v TX MAC.	43
Tabuľka 5-4: Tabuľka výstupov pre stavový automat v TX MAC.	44
Tabuľka 5-5: Prehľad výsledkov syntézy.	47
Tabuľka 6-1: Inštrukcie v test procedúre tp_preamble.	49
Tabuľka 6-2: Inštrukcie v test procedúre tp_express.....	50
Tabuľka 6-3: Inštrukcie v test procedúre tp_no_preemption.....	51
Tabuľka 6-4: Inštrukcie v test procedúre tp_preemption.....	53
Tabuľka 6-5: Inštrukcie v test procedúre tp_preemption.....	56
Tabuľka 6-6: Inštrukcie v test procedúre tp_back_to_back.....	57
Tabuľka 6-7: Inštrukcie v test procedúre tp_long_pckt.....	58
Tabuľka 6-8: Inštrukcie v test procedúre tp_min_frag_size.....	59

ÚVOD

V súčasnosti je pozornosť venovaná vývoju protokolov IEEE pod názvom TSN (Time-Sensitive Networking). V praxi sa už používa veľké množstvo protokolov zo skupiny TSN, ale mnohé z nich sú stále vo vývoji a nie sú implementované v komerčných zariadeniach. Protokoly skupiny TSN zaisťujú deterministický prenos dát prostredníctvom sietí IEEE 802. Toto je umožnené protokolmi zaisťujúcim transport s nízkou latenciou, časovou determinističnosťou správ a s predchádzaním strát paketov. Tieto siete môžu nájsť využitie v aplikáciách z odvetví ako aerospace, automotive, industrial a IoT čo sú dynamicky sa rozvíjajúce obory.

Hlavným cieľom tejto diplomovej práce je analyzovať niektoré z podštandardov IEEE 802.1Q a vybrať si jeden z nich, pre implementáciu v jazyku VHDL. V tejto práci sú hlbšie analyzované dva z nich, pretože sa predpokladá ich spolupráca, podľa štandardu 802.1Q v riešeníach prioritizácie pre TSN. Prvým z analyzovaných podštandardov je 802.1QBV vylepšené plánovanie sieťovej prevádzky. Ďalší analyzovaný podštandard je 802.1QBU, ktorý slúži k popisu správneho fungovania pri preempcii rámcov. Tieto podštandardy sú opísané v kapitole dva, spolu s ďalšími časťami štandardu 802.1Q a ďalšími štandardami ktoré sú potrebné pre správne fungovanie sietí TSN. Porovnaním ich vlastností sa zaoberá štvrtá kapitola. V tejto kapitole sú porovnané teoretické hodnoty dátovej priepustnosti a latencie časovo kritických rámcov spolu s vyhodnotením výsledkov.

Základné teoretické znalosti zo sietí, na ktorých je postavený cieľ tejto diplomovej práce, sú popísané v kapitole jedna.

Pre implementáciu bol zvolený štandardný 802.1QBU. Kódovanie a overovanie správnosti komunikácie bolo prevzaté zo štandardu 802.3BR. Tento štandard je alternatíva k štandardnému ethernetu. Tento štandard je popísaný v kapitole tri.

Dizajn navrhovaného IP jadra je popísaný v piatej kapitole. Sú v nej popísané stavové automaty a funkcie, ktoré dizajn vykonáva. Na konci tejto kapitoly sú tiež uvedené výsledky zo syntézy, ktoré podávajú informácie o možnosti nahrania dizajnu do zvoleného FPGA.

Pre overenie správnosti fungovania dizajnu boli navrhnuté testovacie prostredia, ktoré sú riadené test procedúrami. Tieto testovacie prostredia a test procedúry sú detailne popísané v ôsmej kapitole. Každá test procedúra je popísaná sledom inštrukcií, s časom kedy by mali byť vykonané. Je v nej možné nájsť schematické a funkčné zapojenie vo vnútri testovacieho prostredia. Pre každé testovacie prostredie sú v nasledujúcich podkapitolách popísané test procedúry, ktoré ovládajú dané testovacie prostredie.

1 SIEŤOVÁ KOMUNIKÁCIA

V tejto kapitole je stručne popísaný princíp sieťovej komunikácie, za pomoci referenčného modelu OSI. Ďalej tu je popísané ako možno zistiť teoretickú priepustnosť rôznych komunikačných protokolov a optimálnu dĺžku rámca pri dnešných technológiách.

1.1 Model OSI (Open systems interconnection model)

Model OSI (Open System Interconnection) je nástroj pre lepšie pochopenie dátovej komunikácie medzi dvomi sieťovými systémami. Tento model je rozdelený do siedmych vrstiev. Každá z vrstiev má špecifickú funkciu pre vrstvu nad ňou a ponúka servisné funkcie pre vrstvu pod ňou. Tri najspodnejšie vrstvy sú zamerané na prepúšťanie prevádzku prostredníctvom siete do koncového systému. Vrchné štyri vrstvy sú použité v koncovom systéme pre kompletizovanie komunikačného procesu.[1]

Sieťový model ponúka všeobecné prostriedky na oddelenie funkcií počítačových sietí do viacerých vrstiev. Takýto model vrstvených funkcií je tiež nazývaný "protokolový zásobník" alebo "protokolový balíček".[1]

Protokoly alebo pravidlá môžu fungovať v hardvéri, softvéri, alebo rovnako ako pri väčšine protokolových balíčkov v kombinácii týchto dvoch. Charakter týchto kombinácií je, že spodné vrstvy vykonávajú svoju prácu pomocou hardvéru alebo firmvéru (softvér, ktorý beží na špecifickom hardvéri), zatiaľ čo vyššie vrstvy spracováva softvér.[1]

Tento model je definovaný normou ISO (Medzinárodná organizácia pre normalizáciu) štandard 7498-1, ktorý umožňuje fungovanie všetkých prvkov siete spoločne, bez ohľadu na to, kto vytvoril protokoly a aký počítačový predajca ich podporuje.[1]

1.1.1 Fyzická vrstva

Fyzická vrstva je umiestnená na spodku modelu OSI a je navrhnutá na prenos bitových prúdov pomocou elektrických signálov, svetiel, rádiových prenosov. [2] Táto vrstva tiež definuje konektor, špecifikácie rozhrania a požiadavky na médium (kábel). Elektrické, mechanické, funkčné a procedurálne špecifikácie sú poskytované na odosielanie bitových tokov v počítačovej sieti. [1]

1.1.2 Linková vrstva

Linková vrstva definuje komunikáciu na úrovni spojenia, ktorá je nezávislá od média. Toto zariadenie je postavené na médiu závislom na fyzickom kanáli, ktorý poskytuje fyzická vrstva [3]. Vrstva dátového spojenia pozostáva skoro vždy z dvoch podvrstiev, z ktorých najdôležitejšia je Media Access Control (MAC) [3].

MAC prijíma a prenáša kompatibilné rámce. Vytvára a riadi sekvenciu kontroly rámcov (FCS), ktorá je vytvorená a riadená cyklickou kontrolou redundancie (CRC) a zahadzuje deformované rámce. Taktiež pridáva a odstraňuje preambulu a oddeľovač začiatku rámcu (SFD). [4]

Existujú tri typy ethernetového rámca, ale všetky majú rovnaký formát, len sú rôznej dĺžky:

- Každý rámec začína s preambulou, sedem bajtových polí, ktoré majú rovnakú hodnotu 0x55, platí to pre všetky typy rámcov. [4]
- SFD je nasledujúci bajt v rámci, pri štandardných rámcoch má vždy hodnotu 0xD5. [4]
- Hlavička rámca sa skladá z dvoch šesť bajtových polí, MAC cieľová adresa a MAC zdrojová adresa, ďalej tam môže byť pridané štvôr bajtové pole štítok (VLAN tag), EtherType je ďalšia dvoj bajtová časť rámcu. Používa sa na označenie protokolu zapuzdreného v užitočných dátach rámca. Rovnaké pole sa tiež môže použiť na označenie veľkosti niektorých ethernetových rámcov. [4]
- Užitočné zaťaženie je pole vyplnené užitočnými dátami, jeho dĺžka je od 42/46 bajtov po 1500 bajtov, ak je menej bajtov ako minimum, MAC pridá zostávajúce bajty, nazývané výplň, tieto bajty majú vždy hodnotu 0x00. [4]
- FCS je štvôr bajtové pole, ktoré pomáha zisťovať poškodenie pomocou výpočtu CRC. CRC sa vypočítava zo zdrojovej a cieľovej adresy, poľa dĺžky / typu a užitočného zaťaženia. [4]

1.1.3 Sieťová vrstva

Pri predávaní dát transportnou vrstvou, sieťovej vrstve sa k týmto dátam na odoslanie pripojí len informácia, kto je koncový príjemca. Pre každý dátový blok (paket) sieťová vrstva vyberá trasu, cez ktorú bude odoslaná. Následne predá paket linkovej vrstve s informáciami o zvolenej trase. Táto funkcia sa nazýva smerovanie (routing). Pre správne fungovanie sieťová vrstva potrebuje základné informácie o topológii siete. Ďalšou dôležitou funkciou je predchádzanie preťaženiu a zahlteniu časti siete, čiže riadenie toku dát čo najrovnomernejšie využitie všetkých prenosových prostriedkov a kapacít. [5]

1.1.4 Transportná vrstva

Transportná vrstva poskytuje vyšším vrstvám kvalitnejšie prenosové služby, než tie, čo sú poskytované sieťovou vrstvou. Vďaka tomu môžu užívatelia alebo vyššie vrstvy odmietť od konkrétnych špecifikácií používanej komunikačnej podsiete. Takže transportná vrstva je rozhraním medzi poskytovateľom prenosových služieb a ich užívateľmi, čo sú tri najvyššie vrstvy. [6]

1.1.5 Relačná vrstva

Relačná vrstva poskytuje mechanizmy pre otváranie, zatváranie a riadenie relácií medzi aplikačnými procesmi koncových užívateľov. Komunikácia relácií pozostáva z požiadaviek a odpovedí, ktoré sa vyskytujú medzi aplikáciami. Relačná vrstva túto požiadavku zaisťuje obvykle pomocou mechanizmu predávania poverenia k prenosu dát. Ďalej ponúka prostriedky, pomocou ktorých ide poverenie predať alebo si ho naopak vyžiadať.[7]

1.1.6 Prezentačná vrstva

V praxi rôzne počítače často používajú vlastnú vnútornú reprezentáciu dát. Pre korektné predávanie dát, medzi počítačmi s rôznou vnútornou reprezentáciou dát, sa musí zaistiť konverzia dát, ktorú má na starosti prezentačná vrstva.[8]

1.1.7 Aplikačná vrstva

Hlavnou funkciou aplikačnej vrstvy je poskytovať prístup aplikácie do prostredia siete na bázy ISO/OSI modelu. Ďalej poskytuje protokoly ako vzdialené prihlásenie k vzdialenému počítaču, prenos súborov, transport elektronickej pošty, podpora sietí, inicializácia vzdialeného počítača a ďalšie.[9]

1.2 802.1

Skupina IEEE 802.1 sa zaoberá rozvojom štandardov a doporučených postupov v oblastiach architektúry, prepájaním, bezpečnosťou a celkovou správou 802 lokálnych a metropolitných sietí (LAN a MAN). Ďalej sa zaoberá protokolmi nad MAC vrstvou.[10]

1.2.1 802.1Q

Štandard IEEE 802.1Q umožňuje prepojenie lokálnych sietí cez MAC premostenia alebo premostením cez virtuálnu lokálnu sieť (VLAN). VLAN uľahčujú spravovanie logických skupín staníc. Stanice v tej istej VLAN komunikujú spolu, ako keby boli zapojené v tej istej LAN, aj keď prenos medzi VLAN je obmedzený. Riadenie VLAN premostení a staníc umožňuje pridať, odstrániť alebo presunúť stanice medzi VLAN.[11]

Tento štandard rozširuje špecifikáciu premostení VLAN tak, aby bolo umožnené poskytovateľovi organizačných služieb používať spoločnú infraštruktúru premostení a sietí LAN. To poskytuje nezávislým zákazníckym usporiadaniam ekvivalent oddelených sietí LAN, premostení a virtuálnych premostených sietí.[11]

Tiež špecifikuje protokoly a protokolové entity v architektúre premostení, čo poskytuje schopnosti detekcie, verifikácie a izolácie prepojenia chýb v rámci premostených sietí. Tieto schopnosti je možné použiť v sieťach pracujúcich s mnohonásobnými nezávislými usporiadaniami, každé s obmedzenou správou prístupu pre ďalšie zariadenie.[11]

1.3 802.3

Medzinárodný štandard pre lokálne a metropolitné siete (LAN a MAN), ktoré používajú CSMA/CD ako metódu pre zdieľanie prístupu k médiu a protokolu IEEE 802.3 (ethernet). Ďalej popisuje formát rámca pre dátovú komunikáciu. Táto medzinárodná norma tiež zahŕňa niekoľko typov médií a metód pre rôzne rýchlosti prenosu dát.[4]

1.4 Efektívnosť prenosu (dátové komunikácie a vytváranie sietí)

Jedným z cieľov dátovej komunikačnej siete je presunúť čo najvyšší objem presných informácií prostredníctvom siete. Čím vyšší je objem dát, tým väčšia je výsledná efektívnosť siete. Účinnosť siete je ovplyvnená charakteristikami obvodov, ako sú chybovosť a maximálna prenosová rýchlosť, rovnako ako rýchlosť pre vysielanie a príjem zariadení, ich detekcia chýb a kontrolné metódy a protokoly používané na linkovacej vrstve.[12]

Každý protokol používa niektoré bity alebo bajty na vymedzenie začiatku a konca každej správy a na kontrolu chýb. Tieto bity a bajty sú nevyhnutné na vysielanie, ale nie sú súčasťou správy. Nedodávajú používateľovi informačnú hodnotu, ale počítajú sa proti celkovému počtu bitov, ktoré sa môžu prenášať. [12]

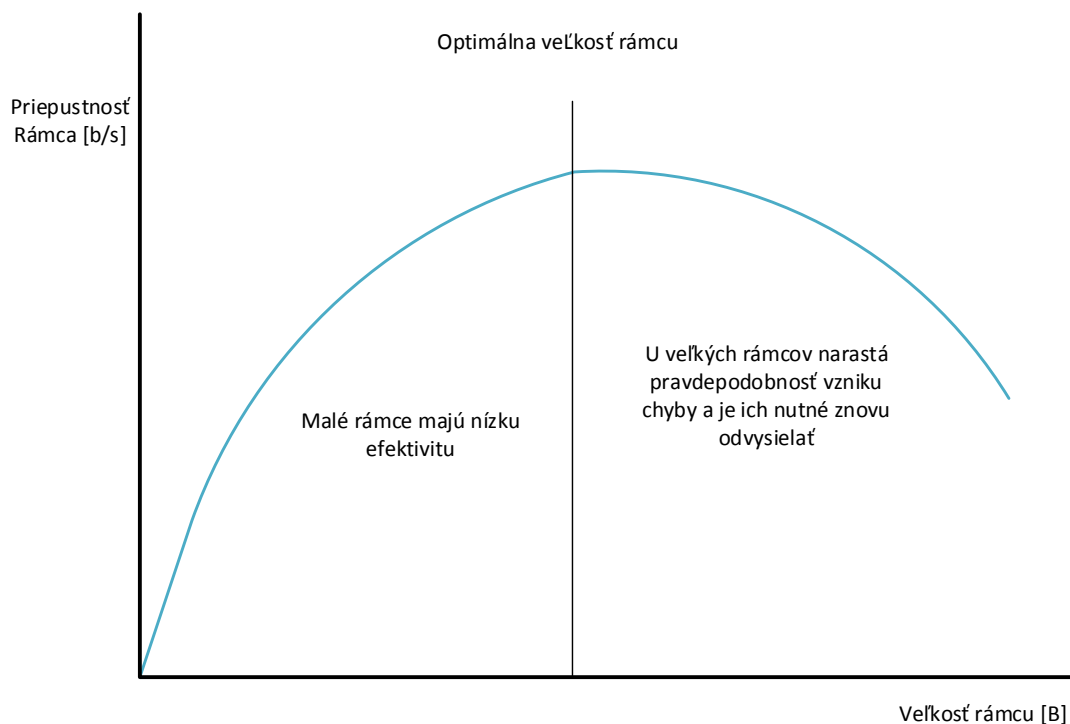
Každý komunikačný protokol má informačné bity a prídavné bity. Informačné bity sú tie, ktoré slúžia na vyjadrenie informácie. Prídavné bity sa používajú na účely ako je kontrola chýb a označenie začiatku a konca znakov a paketov. Bit parity, ktorý sa používa na kontrolu chýb, je prídavný bit, pretože sa nepoužíva na prenos informácií. [12]

Efektívnosť prenosu je definovaná ako podiel celkového počtu bitov informácií (bity odoslané v správe) s celkovým počtom bitov v prenosu (informačné bity plus prídavné bity). [12]

Rovnaký základný vzorec sa môže použiť na výpočet účinnosti synchronného prenosu. [12]

Všeobecným pravidlom je, že čím väčšie je pole správy, tým je protokol efektívnejší, ale ak sa kedykoľvek prijme rámec obsahujúci chybu, celý rámec sa zahodí a musí byť znovu vyslaný, preto sa nepoužívajú štandardné rámce väčšie ako 1536 bajtov. Pravdepodobnosť, že rámec obsahuje chybu, sa zvyšuje s veľkosťou rámca, väčšie rámce pravdepodobnejšie obsahujú chyby. Menšie rámce sa používajú kvôli zákonu pravdepodobnosti. [12]

Na Obr.1.1 je zobrazený graf závislosti priepustnosti dátovej komunikácie na dĺžke vysielaného rámca. [12]



Obr. 1.1: Graf závislosti priepustnosti na veľkosti rámcu. [12]

Priepustnosť je celkový počet bitov informácií prijatých za sekundu po zohľadnení pridaných bitov a potreby opakovaného vysielania rámcov obsahujúcich chyby. Všeobecne povedané, malé rámce poskytujú lepšiu priepustnosť pre komunikáciu s viacerými chybami, zatiaľ čo väčšie rámce poskytujú lepšiu priepustnosť v sieťach menej náchylných na chyby. Vo väčšine reálnych sietí je krivka na vrchu veľmi plochá znázornená na Obr.1.1, čo znamená, že existuje veľa rozmerov rámcov, ktoré poskytujú takmer optimálny výkon. Veľkosti rámcov sa medzi jednotlivými sieťami značne líšia, ale ideálna veľkosť rámu sa pohybuje medzi 2000 a 10 000 bajtmi. [12]

Optimalizácia výkonu v sieti, najmä v sieti klient-server, môže byť zložitá, pretože význam veľkosti rámcu vie veľmi ovplyvniť rýchlosť komunikácie. Výber správnej alebo nesprávnej veľkosti rámcu môže mať väčší vplyv na výkon ako čokoľvek, čo by ste mohli robiť na serveri. [12]

Ideálna veľkosť rámcu závisí od konkrétnej aplikácie a vzoru správ, ktoré generuje. Pre štandardnú komerčnú veľkosť je ideálna veľkosť rámcu pohybujúca sa medzi 4000 a 8000 bajtmi. Bohužiaľ, nie všetky sieťové softvérové balíky umožňujú manažérom siete jemne doladiť veľkosti rámcu týmto spôsobom, ale štandardizovaná veľkosť rámcu je iba 1536 bajtov a to z toho dôvodu, že Ethernet bol štandardizovaný mnoho rokov dozadu, vtedy bola pravdepodobnosť chyby vyššia. Rámce Jumbo a super Jumbo vznikli na optických sieťach s vysokou rýchlosťou prenosu a veľmi nízkou pravdepodobnosťou výskytu chyby. [12]

Výpočet skutočnej výkonnosti dátovej komunikačnej siete je zložitý, pretože závisí nielen od efektívnosti protokolu dátového spojenia, ale aj od miery chybovosti a počtu opakovaných prenosov. Rýchlosť vysielania informačných bitov (TRIB) je mierou

účinného počtu informačných bitov, ktoré sa prenášajú cez komunikačný obvod za jednotku času. Základná rovnica TRIB z ANSI je definovaná vzťahom číslo 1.1. [12]

$$TRIB = \frac{K(M - C)(1 - P)}{\left(\frac{M}{R}\right) + T} \quad (1.1)$$

Kde:	K	je počet informačných bitov na jeden znak
	M	je dĺžka rámcu (počet znakov v rámci)
	R	je rýchlosť prenosu údajov v počte znakov za sekundu
	C	je priemerný počet neinformačných znakov na jeden rámec
	P	je pravdepodobnosť, že rámec bude potrebovať znovu odoslanie z dôvodu výskytu chyby
	T	je čas medzi rámcami v sekundách

2 SIETE TSN (TIME-SENSITIVE NETWORKING)

TSN je technologický štandard, definovaný v IEEE 802.1Q, ktorý poskytuje deterministickú komunikáciu na štandardnom ethernet. TSN technológia je centrálne riadená, využíva časové plánovanie pre aplikácie v reálnom čase, ktoré vyžadujú determiničnosť. Týmto minimalizuje jitter (časový rozptyl medzi dátovými rámcami). Uvedená technológia sa zaoberá druhou vrstvou OSI modelu, TSN je štandard ethernetu a nie internet protokol štandard.[13]

Deterministická komunikácia je dôležitá pre rôzne odvetvia v priemysle napríklad aerospace, automotive, výroba a preprava, pričom poskytovanie prostriedkov pre determinizmus cez štandardný ethernet umožňuje lepšiu konektivitu a optimalizáciu. [13]

Tieto siete využívajú VLAN (virtual local area network) ide o logické a nezávislé siete v rámci jedného alebo niekoľkých zariadení. Za pomoci týchto sietí je možné vytvoriť logickú organizáciu nezávislú na fyzickej vrstve, tým sa uľahčí správa siete, zvýši výkon a zlepši bezpečnosť. Obvykle sa realizuje na komponentoch nazývaných prepínač. Porty tohto prepínača sú rozdelené na niekoľko logických samostatných častí, jedná sa o delenie sietí na druhej vrstve podľa modelu ISO/OSI. [13]

V tejto práci sú popísané hlavne časti štandardu, ktoré sa zaoberajú prioritizáciou rámcov pri vysielaní. Ako prvá je popísaná časť štandardu 802.1QBV, táto časť sa zaoberá vylepšeným plánovaním vysielania rámcov. Druhou popisovanou časťou štandardu je 802.1QBU, ktorá sa zaoberá preempciou rámcov, prerušením vysielania rámcov s nižšou prioritou a začiatkom vysielania rámcov s vyššou prioritou.

Ďalšou skupinou častí TSN štandardu sú časti, ktoré sa zaoberajú sieťovou konfiguráciou.

Ďalej sa tieto siete zaoberajú časovou synchronizáciou. Pre definovanie časovej synchronizácie sa používajú časti štandardov, ktoré patria do iných skupín. Medzi štandardy ktoré sa používajú patria napríklad IEEE 1588 a IEEE 802.1AS.

2.1 Časová synchronizácia

Pre dosiahnutie poskytovania determiničnosti je dôležité zdieľanie času. Ako príklad pre distribúciu synchronizovanej časovej domény možno použiť synchronizáciu so satelitom GPS (global positioning system), kde synchronizáciu s časovou doménou serveru pre NTP (network time protocol) zaobstaráva synchronizačný kontrolér pre vnútornú časovú doménu v satelite. Ako ďalší príklad sa dajú použiť synchronizačné kontroléry pracujúce so spoločnou časovou doménou, ktorá využíva IEEE 1588, čiže PTP (precision time protocol).[14]

Štandard IEEE 1588 poskytuje mechanizmus pre synchronizáciu časových domén prepojených cez multicast siete, zahrňujúce siete na báze ethernetu. Tento štandard bol navrhnutý tak, aby poskytoval synchronizáciu odolnú voči chybám medzi heterogénnymi sieťovými časovými doménami. Tento štandard tiež definuje PTP. [14]

Ďalší štandard zaoberajúci sa časovou synchronizáciou je IEEE 802.1AS, ktorý bol

vyvinutý pracovnou skupinou pre premostenie audia a videa na optimalizáciu distribúcie časovej synchronizácie v profesionálnych audio/video systémoch. Toto je definované ako interoperabilný mechanizmus pre synchronizáciu času v súvislosti so štandardným ethernetom. Uvedený štandard špecifikuje formálny profil pre IEEE 1588 ako profil pre 1588 konfiguruje a vyberá z dostupných možností z 1588, z ktorých sú zahrnuté nasledujúce: [14]

- Použitie modifikovaného BMCA (best master clock algorithm), umožnenie rýchlej konvergenencie na časovej doméne grandmaster.
- Použitie mechanizmu oneskorenia peer na meranie oneskorenia sieťovej cesty.
- Transport správy pre časovú synchronizáciu cez ethernet a koordináciu siete zdieľaných sietí.
- Direktívna účasť mostov vrátane účasti pri výbere najlepšej hodinovej domény.
- Zahrnutie merania najbližšieho frekvenčného posunu v koncových staniciach a mostoch pre lepšiu synchronizáciu presnosti a rýchlosti konvergenencie grandmaster-a.
- Špecifikácia požiadaviek na výkon tak, že akékoľvek dva časovo známe systémy oddelené siedmimi alebo menej preskokmi budú synchronizované do 1 μ s navzájom.

2.2 Sieťová konfigurácia

Základnými elementmi systémov na báze TSN a audio/video premostovania sú časová synchronizácia a tvarovanie sieťovej prevádzky, takže tieto systémy potrebujú dynamickú konfiguráciu založenú na požiadavkách aplikácii. V tejto kapitole sú popísané konfiguračné mechanizmy, vysvetľujúce ako plne centralizovaný, konfiguračný model je optimalizovaný pre použitie v sieťach TSN, použitých v kontrolných aplikáciách. [14]

Časť TSN štandardu 802.1QAT bola vyvinutá ako rezervačný mechanizmus šírky pásma pre audio/video premostovacie systémy, ktoré využívajú tvarovač pre sieťovú prevádzku založený na kreditnom systéme, ktorý je popísaný v časti štandardu 802.1QAV (Presmerovanie a radenie do frontu pre časovo kritické dátové toky). 802.1QAT definuje protokol pre rezerváciu dátového toku, plug-and-play konfiguračný mechanizmus nastavenia a odtrhnutia rezervácie toku dát. [14]

Použitím protokolu pre rezerváciu dátového toku, zdroje vysielania deklarujú požiadavku na šírku pásma pred začiatkom prenosu dátového toku. Tieto požiadavky na šírku pásma sú predané sieti za pomoci inzerčných správ zdroja vysielania. Tieto správy definujú servisné požiadavky na kvalitu dátového toku, čo zahŕňa triedu sieťovej prevádzky, rýchlosť prenosu dát a najhorší prípad akumulovanej latencie. Tieto správy sú propagované cez premostenú sieť až k potencionálnemu prijímateľovi. Akumulovanú latenciu sa prijímateľ dozvie zo správ od vysielateľa, latencia sa aktualizuje pri každom preskoku, takže prijímateľ sa dozvie najhoršiu možnú latenciu. Ak premostenie nepodporuje požiadavku na šírku pásma pre tok dát, potom pošle odosielateľovi správu

o zlyhaní a rezervácia toku dát zlyhá. [14]

Prijímateľ, ktorí prijíma vysielateľove inzerčné správy, odpovedá, že chce prijať tok dát odoslaním správy vysielateľovi, že je pripravený. Ako sa táto správa propaguje cez premostenú sieť, premostenia uzamknú zdroje potrebné pre dodanie toku dát v súlade s požiadavkami na kvalitu služby. Akonáhle vysielateľ obdrží prijímateľovu správu, že je pripravený, môže začať prenos toku dát. [14]

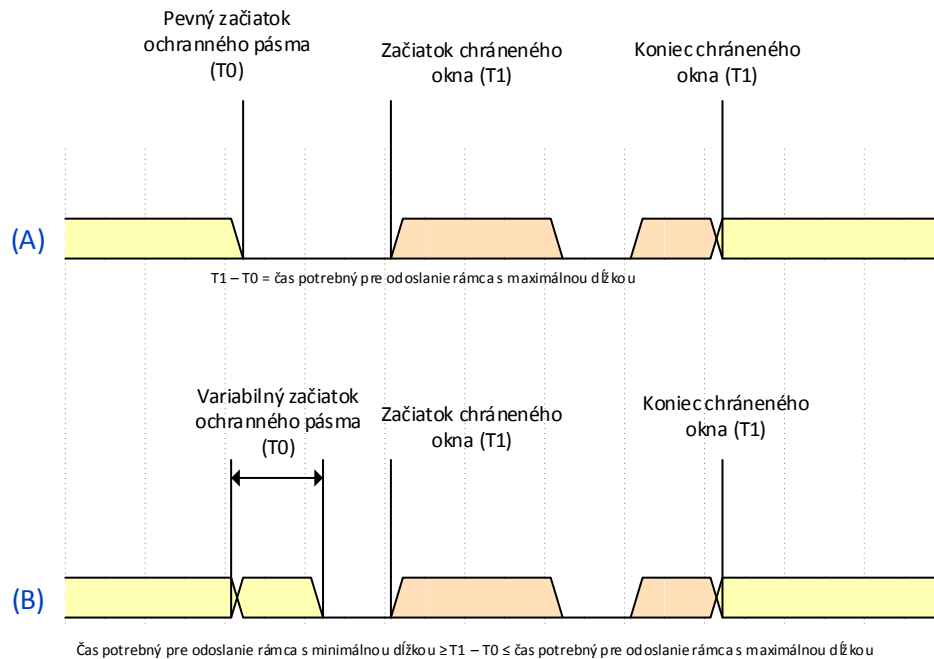
Časť štandardu TSN 802.1QCC poskytuje zlepšenia existujúceho protokolu pre rezerváciu toku dát. Tieto zlepšenia poskytujú podporu pre viac tokov dát, konfigurovanie tried pre rezerváciu toku dát a tried toku dát. Ďalej poskytujú lepší popis charakteristiky toku dát, možnosť prenosu toku dát cez tretiu vrstvu OSI modelu, deterministickú konvergenciu rezervácie toku dát a užívateľské sieťové rozhranie pre smerovanie a rezerváciu. TSN konfigurácia sa môže dosiahnuť staticky navrhovateľom alebo dynamicky prostredníctvom sieťovej služby. [14]

2.3 802.1QBV

Štandard pre lokálne a metropolitné siete - mosty a virtuálne premostované lokálne siete. Modifikácia zlepšenia plánovanej prevádzky. [15]

V tomto dodatku sú špecifikované procedúry na vyprázdňovanie radov podľa časovania rozmeru. Plánovanie prenosu rámcov na základe časovania odvodeného zo štandardu IEEE 802.1AS, ktoré je povolené riadenými objektmi a rozšíreniami existujúcich protokolov. Štítiky virtuálnej lokálnej siete (VLAN) sú zakódované hodnoty, priority umožňujúce simultánnu naplánovanú prevádzku, kreditovo založené tvarovanie prevádzky a iný premostený prenos cez lokálne siete (LAN). Použitie tohto dodatku prináša lepšie predvídateľné doručovanie rámcov v čase, v ktorom prenos rámcov začne a celkové oneskorenie s časovým nepokojom, ktoré sa vyskytne v mieste určenia. Toto umožní zmiešať časovo kritickú prevádzku s inými triedami sieťovej prevádzky v tej istej sieti, ak toto miešanie spĺňa časové požiadavky časovo kritickej sieťovej prevádzky. Vytvorenie priorít je nedostačujúce, preto musí byť zabezpečené, že rámce s nižšou prioritou majú dostatok času na dokončenie vysielania pred začiatkom času vyhradeného pre prenos kritických rámcov, inak by mohlo dôjsť k oneskoreniu vysielania až o čas potrebný pre prenos rámcu s maximálnou dĺžkou. [15]

Pre implementáciu je potrebné zabezpečiť, aby prístup k vysielacím rámcom mala iba jedna komunikačná trieda, alebo skupina tried čo vytvára chránený kanál pre túto triedu, iný prenos musí byť zastavený s dostatočnou časovou rezervou pred začiatkom chráneného kanálu. Prenos v nechránenom kanále môže začať iba ak je čas do začiatku chráneného okna väčší alebo rovný času, ktorý je potrebný na prenos rámca s maximálnou dĺžkou. Týmto sa vytvára ochranné pásmo pred časom určeným pre chránenú prevádzku. V tomto ochrannom pásme nemôže začať vysielanie žiadneho rámca. Šírka ochranného pásma môže byť nastavená na časovú dĺžku potrebnú na prenos rámca s maximálnou dĺžkou, alebo učením veľkosti ďalšieho rámca v rade na prenos a ak je dostatok času na prenos rámca s touto dĺžkou, je možné začať prenos tohto rámca. Obidva riešenia sú zobrazené na Obr. 2.1.[15]

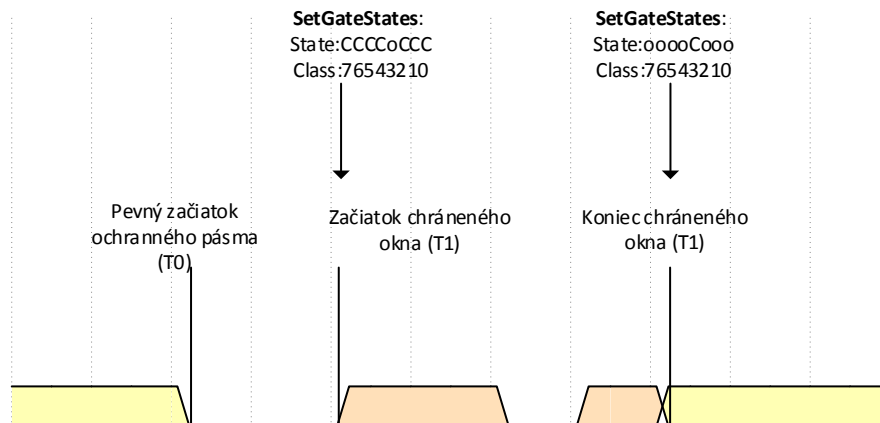


Obr. 2.1: Ochranné pásmo A) pevný začiatok ochranného pásma B) premenný začiatok ochranného pásma.

2.3.1 Vytváranie chráneného okna, používaním operácií na bránach

Vylepšenia implementácie pre plánovanú prevádzku umožňuje zapnutie a vypnutie prenosu načasovanú pre všetky triedy prevádzky na príslušnom porte. Všetky triedy sieťovej prevádzky majú individuálne spínacie prenosové brány, ktoré sú riadené zoznamom operácií pre brány, tento zoznam môže byť programovateľný. Funkcia `SetGateStates` funguje s parametrom časového oneskorenia, ktorý je časom medzi dvoma po sebe nasledujúcimi operáciami a parametrom `GateState`, čo je osem bitový vektor, ktorý definuje stav každej brány pre celé triedy sieťovej prevádzky. Táto operácia môže vytvoriť akúkoľvek kombináciu stavu brán a chránených tried. Ukončí sa na konci zoznamu a reštartuje sa po uplynutí `OperCycleTime` od začiatku cyklu. [15]

Obr. 2.2 opisuje, ako môžu operácie brány vytvoriť chránené okno. V čase T_1 sa nastavuje brána pre chránenú triedu sieťovej prevádzky do stavu otvorená a brány pre ostatné triedy sieťovej prevádzky sa nastavujú do stavu zatvorená. V čase T_2 táto operácia zatvorí bránu pre túto triedu a zároveň otvorí brány ostatným triedam. [15]



Obr. 2.2: Príklad operácií prebiehajúcich na bránach.

2.3.2 Presný časový protokol

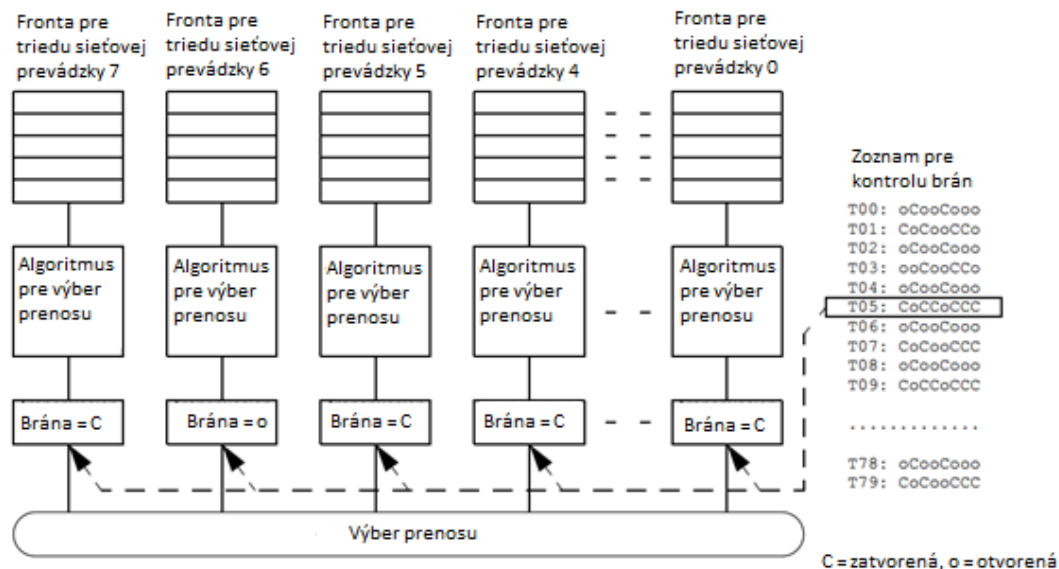
Spustenie protokolu PTP a dostupnosť času z PTP umožňujú lepšie načasovanie operácií brány. Tento protokol nie je potrebný na implementáciu, môže byť nahradený inými alternatívnymi mechanizmami, ktoré zabezpečujú spoločný pohľad na čas. [15]

2.3.3 Plánovaná prevádzka a koncové stanice

Premostenia by mali podporovať vylepšenia pre plánovanú sieťovú prevádzku aj pokiaľ sú koncové stanice nepripojené do týchto premostení. Plánovací mechanizmus nesmie byť implementovaný v koncových staniách. To znamená, že koncové stanice sú len príjemcami plánovanej sieťovej prevádzky. [15]

2.3.4 Prepojenie vo vnútri 802.1QBV

Na Obr. 2.3 je zobrazené prepojenie vo vnútri hardvéru popísaného v tomto podstandarde. Zloženie hardvéru je z frontov pričom každá trieda sieťovej prevádzky má vlastný front, Algoritmus pre selekciu prenosu rozhoduje o tom, ktorá trieda sieťovej prevádzky má prednosť pred inou, v prípade viacerých otvorených brán. Zoznam pre kontrolu brán rozhoduje v ktorom čase, ktoré brány budú otvorené a ktoré budú zatvorené. Časť výber prenosu prepája výstup z brán so vstupom do nižšej podvrstvy v rámci linkovacej vrstvy.



Obr. 2.3: Schematické zobrazenie zapojenia pre podštandard QBV.

2.4 802.1QBU

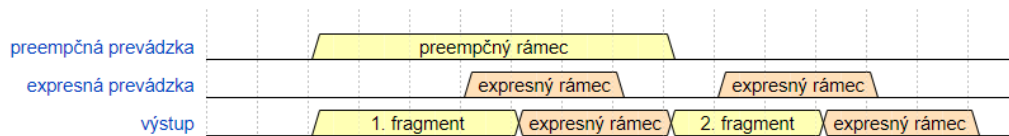
Tento dodatok sa zaoberá preempciou ethernetových rámcov. Ďalej špecifikuje procedúry, spravovanie objektov a rozšírenie protokolov, ktoré sú potrebné pre funkčnú preempciu ethernetových rámcov: [16]

- Definuje triedu služby pre časovo kritické ethernet rámce. Tie vyžadujú, aby vysielateľ v premostenej lokálnej sieti prerušil vysielanie triedy sieťovej prevádzky, ktorá nie je časovo kritická. Tým sa uvoľní sieťová prevádzka pre vysielanie časovo kritických rámcov. Po dokončení prenosu časovo kritických rámcov sa obnoví odosielanie prerušeného best effort rámca, pričom prerušenie best effort rámca, ktorý je časovo nekritický, je umožnené viac krát. [16]
- Zabezpečenie vyhľadávania, konfigurácie a kontroly služby preempcií ethernetových rámcov pre mostové porty a koncové stanice. [16]
- Preempcia ethernetových rámcov je zaistená len vtedy, ak v danom prepojení majú obidva zariadenia podporovanú funkciu preempcie ethernetových rámcov. [16]

Cieľom tohto pozmeňujúceho a dopĺňujúceho dodatku je poskytnúť zníženú latenciu prenosu pre plánované časovo kritické rámce v premostenej sieti LAN. Po tomto pozmeňujúcom a dopĺňujúcom dodatku sa môže spustiť vysielanie rozsiahleho nekritického rámca pred začiatkom vysielania časovo kritického rámca. Bez tohto dodatku vzniká nadmerná latencia pre časovo kritické rámce. Veľké množstvo prenosu bez použitia preempcie ethernetových rámcov, bráni schopnostiam aplikácií, ktoré sa používajú pre naplánovaný prenos ethernetových rámcov na implementáciu riadenia siete

v reálnom čase. [16]

Obr. 2.4 zobrazuje jednoduchú funkciu preempcie ethernetových rámcov. Odtiaľ je možné vidieť, že pokiaľ prebieha prenos preempčného ethernet rámca, ktorý môže byť preemptovaný, a príde na vstup expresný rámec pripravený na vysielanie. Následne sa skontrolujú počet odoslaných bajtov. Ak je počet odoslaných bajtov väčší alebo rovný ako minimálna dĺžka fragmentu a zároveň počet bajtov, ktoré sa ešte musia odoslať, musí byť väčší alebo rovný minimálnej dĺžke fragmentu. Po splnení týchto dvoch podmienok nastane prerušenie vysielania preempčného rámca. Po prerušení preempčného rámca môže začať vysielanie z expresnej triedy sieťovej prevádzky. Po skončení vysielania rámcov z expresnej prevádzky sa obnoví vysielanie prerušeného preempčného rámca.



Obr. 2.4: Príklad preempcie rámcov.

2.4.1 Preempcia použitá v izolácii

Prerušenie prenosu ethernet rámca s nižšou prioritou (preempčný) inými rámcami s vyššou prioritou (expresné) je povolená preempcia. Prenos prerušeného preempčného rámca pokračuje po dokončení prenosu expresných rámcov. Keď je expresný rámec pripravený na vysielanie a preempčný rámec je už prenášaný, potom oneskorenie expresného rámca bude v najhoršom prípade doba potrebná na prenos 123 bajtov. Avšak často to bude doba potrebná na prenos rámca s dĺžkou 64 bajtov alebo rámca s menšou dĺžkou. V prípade používania preempcie bez plánovania, zníži sa tým jitter (časový rozptyl medzi dátovými rámcami) v časovo citlivých rámcoch. Všetky triedy sieťovej prevádzky s rovnakou prioritou by mali mať rovnaké označenie. Pokiaľ expresné a preempčné priority sú zlúčené do rovnakej triedy sieťovej prevádzky, expresný rámec nemôže zabrániť preempčnému rámcu z tejto fronty. [16]

2.4.2 Plánovanie a preempcia použitá v kombinácii bez HOLD/RELEASE

Plánovací mechanizmus je popísaný v časti 2.3, ktorá hovorí o podštandarde 802.1QBV jeho vylepšeniach plánovania a "ochrannom pásme". Prenos by mal byť dokončený pred začiatkom prenosu ďalšej triedy. [16]

Pokiaľ je plánovanie, ktoré sa vzťahuje na preempčné triedy sieťovej prevádzky, navrhnuté tak, umožňovalo prenášať preempčným triedam sieťovej prevádzky (napríklad neexistuje žiadna udalosť ako zavretie brán pre preempčné triedy sieťovej prevádzky). To umožňuje vysielanie preempčného rámca kedykoľvek až do začiatku prenosu expresnej triedy sieťovej prevádzky. V závislosti na načasovaní môže vysielanie preempčnej prevádzky pokračovať až po dobu potrebnú na odvysielanie 123 bajtov po otvorení brány pre expresnú triedu sieťovej prevádzky. To dramaticky znižuje oneskorenie ktoré sa prejavuje v preempčnej sieťovej prevádzke, to však na úkor zvýšenia meškania expresnej sieťovej prevádzky. Aj keď to môže byť prijateľné pre niektoré aplikácie, nie je to však prípustné pre aplikácie kde je nevyhnutné znížiť alebo odstrániť

akékoľvek oneskorenie, ktoré sa môže vyskytnúť pri prenose expresných rámcov. [16]

2.4.3 Plánovanie a preempcia použitá v kombinácii s HOLD/RELEASE

Preempcia ethernet rámcov poskytuje mechanizmus HOLD a RELEASE, tento mechanizmus implementuje explicitné „ochranné pásmo“ okolo chráneného prenosového okna. Požiadavka pre HOLD by mala byť naplánovaná na čas potrebný pre odoslanie 124 bajtov pred začiatkom expresnej prevádzky v chránenom okne, z toho vyplýva, že začiatok vysielania rámca musí byť maximálne v čase potrebnom na odvysielanie 124 bajtov pred začiatkom chráneného okna a vysielanie sa musí skončiť pred začiatkom chráneného okna. [16]

Použitím kombinácie mechanizmu HOLD a RELEASE, plánovania a preempcie rámcov spôsobí, že chránené okno je úplne chránené pred rozhraním z predbežnej prevádzky a minimalizuje časový vplyv na predbežnú prevádzku chráneného okna. [16]

2.4.4 Pridelovanie šírky pásma a expresná prevádzka siete

MAC definovaný v štandarde 802.3 neurčuje, ako dlho môže expresná trieda vykonávať preempciu ethernet rámca. Je to v rozpore s týmito dvomi podmienkami:

- Ak sa vyskytne nejaká preempčná sieťová prevádzka, existujú požiadavky na oneskorenie vysielania mostu. [16]
- Nie je možné usporiadať expresnú prevádzku, aby obsadila celú dostupnú šírku pásma v prípade, že neexistujú neobmedzené fronty. To je opísané teóriou čakania. [16]

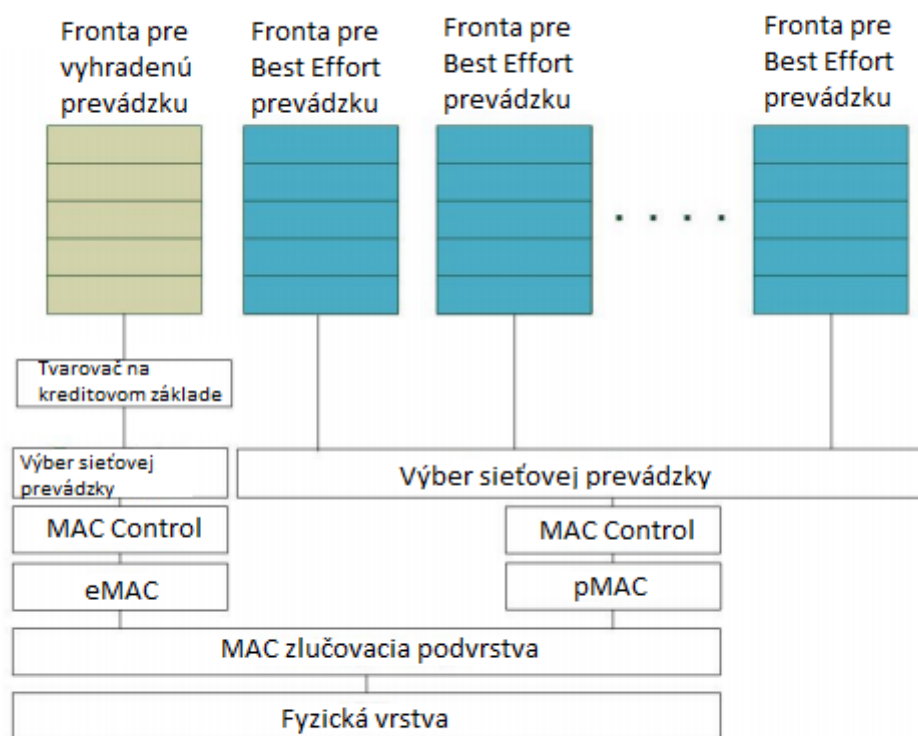
Efektívny mechanizmus preempcie potrebuje rozdelenie šírky pásma menej ako 100% pre expresnú triedu sieťovej prevádzky podľa časového rozvrhu porovnateľného s oneskoreniami tolerovanými časovo citlivou prevádzkou. [16]

Preempcia ethernet rámcov môže spôsobiť, že oneskorenie nebude viditeľné na ochranu pred oneskoreným tranzitom. Toto oneskorenie by malo byť oveľa menšie ako požiadavky pre oneskorený prenos. Oneskorenie sa dá vyriešiť krátkymi rýchlymi dopravnými oknami a krátkymi medzerami medzi nimi, dostatočnými pre čas na maximálne oneskorenie. [16]

Preempcia ethernet rámca bez plánovania je vhodná na použitie a implementáciu v sieti s očakávaním malého objemu expresnej prevádzky. Expresná prevádzka by mala používať tvarovač, ktorý je podobný tvarovaču založenom na kreditoch s podobnou primeranou väzbou na oneskorenie pri vysielaní. [16]

2.4.5 Prepojenie v 802.1QBU

Na Obr. 2.5 je zobrazené prepojenie vo vnútri hardvéru popísaného v tomto podštandarde. Obsahuje Fronty, jeden pre každú triedu sieťovej prevádzky, selekciu prenosu, jeden je určený pre expresnú prevádzku a druhý pre preempčnú prevádzku. Táto časť je presne opísaná dodatkom 802.1QBV, dvoma MAC kontrolami, dvoma MAC-mi, jedným pre každú prevádzku, MAC Merge Sublayer, ktorý spája obe prevádzky do jednej a prepojením na fyzickú vrstvu.



Obr. 2.5: Bloková schéma pre 802.1QBU.

3 802.3BR

Pre kódovanie preempčnej prevádzky sa nepoužíva štandardné kódovanie, ako pri klasickom ethernet, ale kódovanie popísané v štandarde 802.3BR. Tento štandard nie je súčasťou štandardov zo skupiny TSN, ale je alternatívou k štandardnému ethernetu.

3.1 Formát rámcu podľa 802.3BR

Expresné rámce majú formát ako klasické ethernet rámce, sedem bajtov preambula 0x55, jeden bajt SFD 0xD5, šesť bajtov MAC cieľová adresa, šesť bajtov MAC zdrojová adresa, dva bajty ethertype, dátové pole s veľkosťou 46 až 1500 bajtov a na konci FCS s iniciálnou hodnotou 0xFFFFFFFF.[17]

Preempčné rámce, bez fragmentácie sa trochu odlišujú, namiesto SFD sa používa SMD-Ix(start mframe delimiter, initial fragment). [17]

Porovnanie formátu expresných a preempčných rámcov s formátom klasického ethernet rámca je zobrazené na Obr.5.1. Napravo od grafického modelu rámca sú napísané počty bajtov, z ktorých sa skladá daná časť rámca. [17]

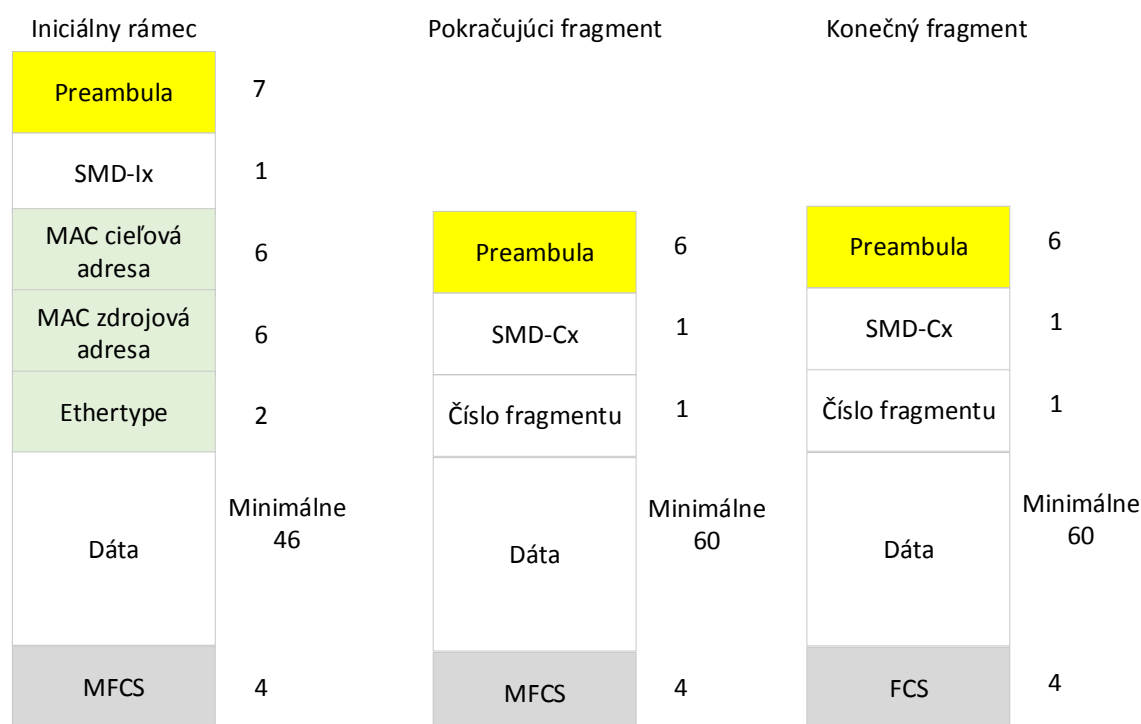
MAC rámec		Expresný rámec		Preemptovateľný rámec	
Preambula	7	Preambula	7	Preambula	7
SFD	1	SFD	1	SMD-Ix	1
MAC cieľová adresa	6	MAC cieľová adresa	6	MAC cieľová adresa	6
MAC zdrojová adresa	6	MAC zdrojová adresa	6	MAC zdrojová adresa	6
Ethertype	2	Ethertype	2	Ethertype	2
Dáta	46 až 1500	Dáta	46 až 1500	Dáta	46 až 1500
FCS	4	FCS	4	FCS	4

Obr. 3.1: Porovnanie formátov klasického MAC rámca, Expresného rámca a Preemptovateľného rámca.

Pri fragmentácii preempčného rámca vyvolanej preempciou, sa môžu fragmentovať len rámce dlhšie ako 128 bajtov bez preamble a SMD a to z dôvodu, že každý fragment musí byť minimálne tak dlhý ako je minimálna dĺžka ethernet rámca bez preamble a SFD, čo je 64 bajtov. Môžu vzniknúť až tri typy fragmentov preempčného rámca: [17]

- Iniciálny fragment, tento fragment je identický s nefragmentovaným preempčným rámcom až na to, že koncová štvorbajtová sekvencia MFCS (mframe check sequence) na rozdiel od FCS má iniciálnu hodnotu 0xFFFF0000. [17]
- Pokročilé fragmenty, tieto fragmenty sa nachádzajú medzi prvým (iniciálnym) fragmentom a posledným fragmentom daného rámca, skladajú sa zo šiestich bajtov preamble, namiesto SMD-Ix majú SMD-Cx (start mframe delimiter continue fragment), číslo fragmentu, dátová časť s minimálnou dĺžkou 60 bajtov a MFCS. [17]
- Posledný fragment, tento fragment začína rovnako ako pokročilé fragmenty, dátová časť, tiež musí byť minimálne 60 bajtov dlhá, ale na konci tohto rámca je FCS celého rámca. [17]

Na Obr.5.2 sú zobrazené všetky tri možnosti fragmentov rámca.



Obr. 3.2: Formáty fragmentov preempčného rámca.

3.2 Kódovanie SMD a čísla fragmentu

SMD-Ix môže nadobúdať štyri hodnoty, ktoré sa neustále cyklija. Po odoslaní celého rámca sa táto hodnota posunie. [17]

SMD-Cx a číslo fragmentu tiež môžu nadobúdať štyri hodnoty, ktoré neustále cyklija po odoslaní fragmentu. [17]

Všetky tieto hodnoty sú zapísané v tabuľke 5.1.

Tabuľka 3-1: Hodnoty pre SMD polia a polia s číslom fragmentu. [17]

Číslo	SMD-Ix	SMD-Cx	Číslo fragmentu
0	0xE6	0x61	0xE6
1	0x4C	0x52	0x4C
2	0x7F	0x9E	0x7F
3	0xB3	0xAD	0xB3

4 TEORETICKÝ VÝPOČET A POROVNANIE PRIEPUSTNOSTI 802.QBU S INÝMI ŠTANDARDMI

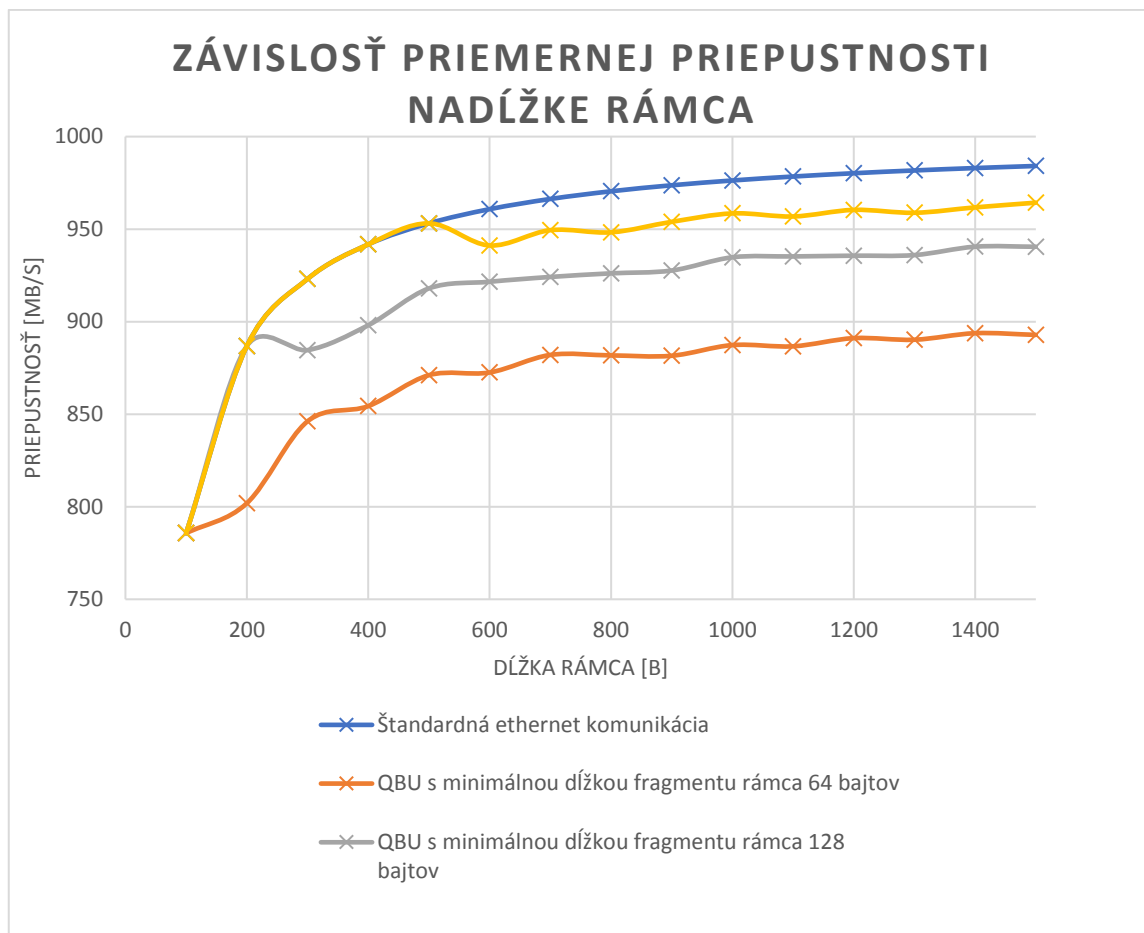
V tejto kapitole sú spočítané teoretické hodnoty priepustnosti navrhovaného štandardu 802.1QBU a iných vybraných štandardov, pre porovnanie s týmto štandardom a priepustnosti. Ďalej je tu spočítaná maximálna latencia tohto štandardu, pri rôznych implementáciách. Priepustnosť bola spočítaná pomocou vzorca 1.1, uvedeného v prvej kapitole. Ako pravdepodobnosť odoslania chybného rámca bola zvolená nulová hodnota, keďže tieto štandardy podporujú maximálnu dĺžku rámca 1500 bajtov a doporučená dĺžka rámca sa v dnešnej dobe pohybuje okolo 8000 bajtov, ako bolo spomenuté v kapitole 1.4, takže pravdepodobnosť odoslania chybného rámca je veľmi nízka a je ju možné zanedbať vo výpočtoch.

4.1 Porovnanie priemernej sieťovej prevádzky cez 802.1QBU a cez štandardný ethernet

Porovnanie 802.1QBU so štandardným ethernetom prebehlo, spočítaním priepustnosti možných implementácií (fragment s minimálnym počtom odoslaných dát 64, 128 a 256 bajtov), výsledná priepustnosť vznikla priemerom dvoch hodnôt, najlepšieho prípadu (maximálne jedna preempcia, jedným 64 bajtovým expresným rámcom, počas vysielania dlhého rámcu) a najhoršieho prípadu (maximálny počet preempcií, jedným 64 bajtovým expresným rámcom, počas vysielania dlhého rámca). Veľkosť dlhého preempčného rámca sa pohybovala od 100 do 1500 bajtov s nárastom hodnoty po 100 bajtoch.

Pre výpočet priepustnosti štandardného ethernetu sa počítalo z celkového počtu bajtov odoslaných štandardom 802.1QBU a počtu nadmerných bajtov pre daný počet rámcov. Výsledná priepustnosť sa tiež spočítala spriemerovaním najlepšieho a najhoršieho prípadu.

Do výpočtov boli ďalej zahrnuté tieto čísla: 8 informačných bitov na jeden znak, rýchlosť prenosu údajov bola stanovená na 125 M znakov za sekundu (1 Gb/s) a čas medzi rámcami bol zvolený na 96 ns (12 znakov medzi jednotlivými rámcami).



Obr. 4.1: Závislosť priemernej priepustnosti na dĺžke rámca, pre štandardný ethernet a varianty podštandardu s rôznou minimálnou dĺžkou fragmentu.

Na Obr.4.1 je zobrazené grafické porovnanie závislosti priepustnosti na veľkosti dlhého rámca, možných implementácií štandardu 802.1QBU s rôznou minimálnou dĺžkou a štandardného ethernetu. U štandardu 802.1QBU veľkosť priepustnosti rastie s minimálnou dĺžkou fragmentu, čo je možné vysvetliť menšou fragmentáciou rámca, čiže menším počtom pridaných znakov v rámci jedného rámca.

4.2 Porovnanie uvažovanej sieťovej prevádzky cez ethernet, 802.1QBU, 802.1QBV a ich kombinácii

Ako uvažovaná sieťová prevádzka, kde na jeden preempčný rámec (dlhý rámec s nižšou prioritou) pripadá jeden expresný rámec (krátky rámec dlhý 64 bajtov), výpočty priepustnosti sa počítali obdobne ako v predchádzajúcej podkapitole.

Pre štandardný ethernet sa preto počítala priepustnosť z dvoch rámcov vysielaných po sebe.

Pre štandard 802.1QBU sa uvažovalo iba s jednou preempciou na preempčný rámec.

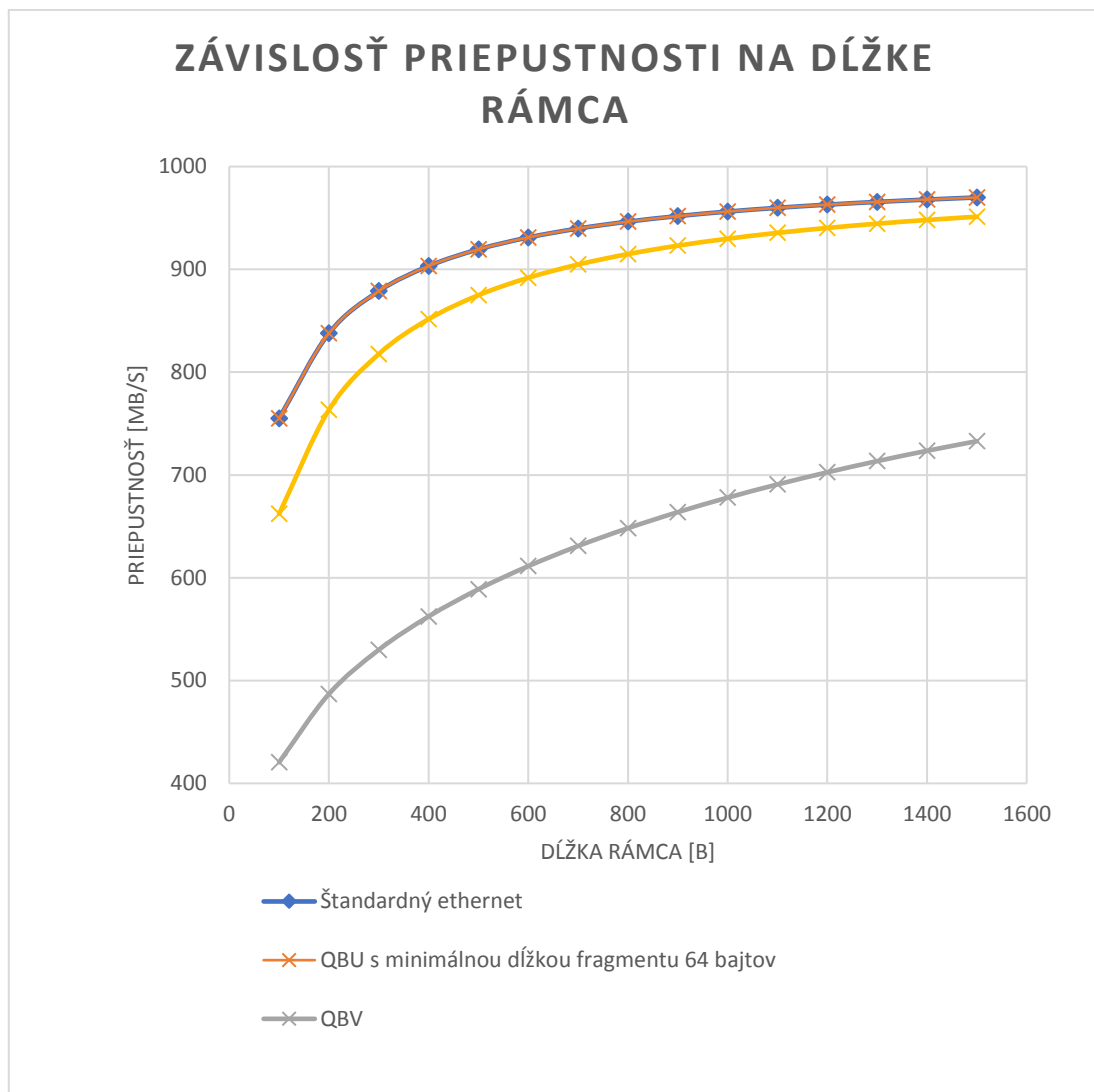
Pre štandard 802.1QBV sa výsledná hodnota priepustnosti počítala ako priemer hodnoty priepustnosti s maximálnou dobou čakania, čo je čas potrebný na odoslanie

rámca s maximálnou dĺžkou (1526 bajtov) a s nulovou dobou čakania, čo je optimálne naplánovaná doba odosielania rámcov.

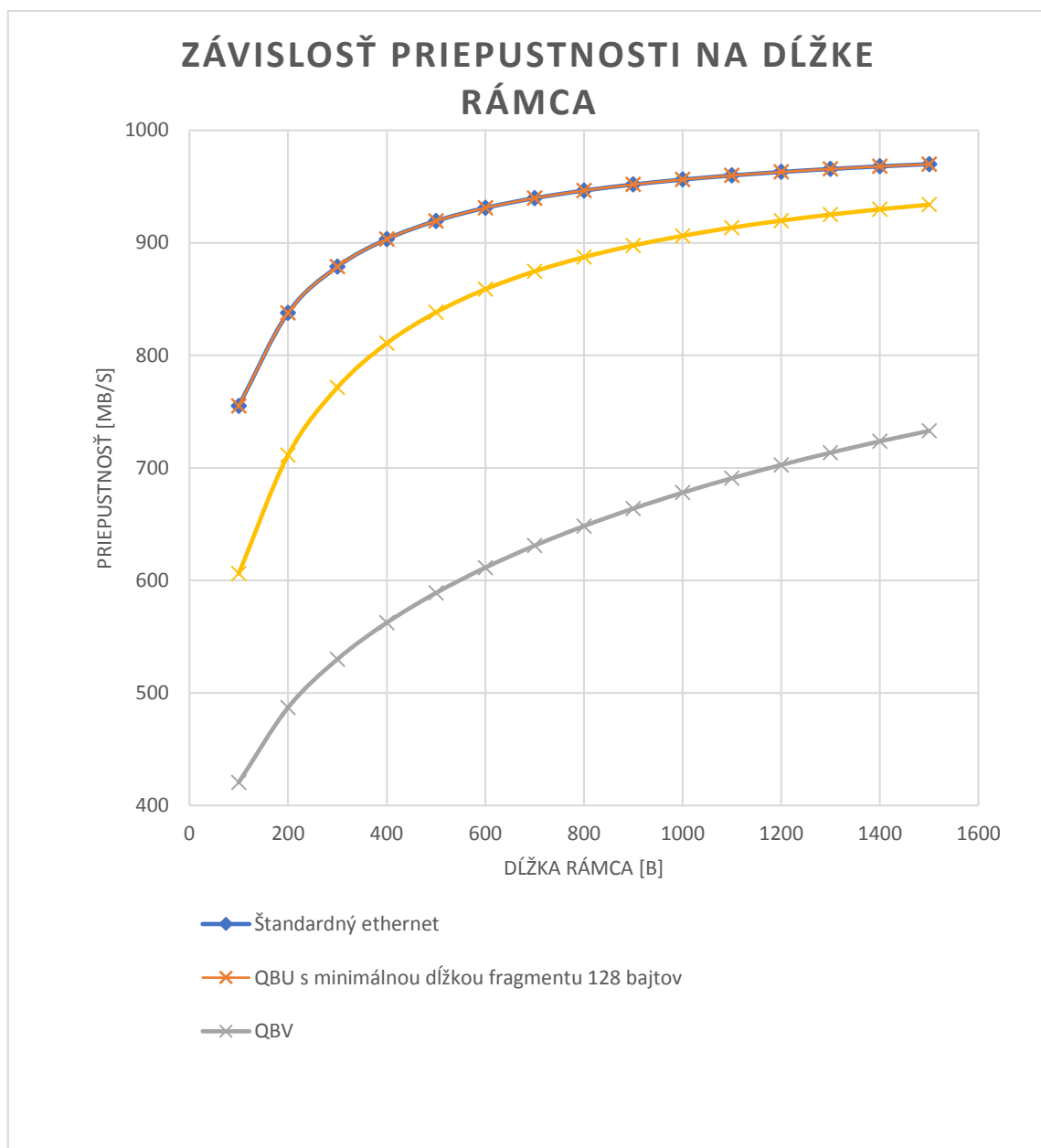
Pri kombinácii štandardov 802.1QBU a 802.1QBV sa výsledná hodnota priepustnosti počítala ako priemer hodnoty priepustnosti s maximálnou dobou čakania, čo je čas potrebný na odoslanie fragmentu rámca s maximálnou dĺžkou (64 až 128 bajtov pre minimálnu dĺžku fragmentu - 64 bajtov, 128 až 256 bajtov pre minimálnu dĺžku fragmentu - 128 bajtov a 256 až 512 bajtov pre minimálnu dĺžku fragmentu - 256 bajtov) a s nulovou dobou čakania, čo je optimálne naplánovaná doba odosielania rámcov.

Obr.4.2 zobrazuje grafické porovnanie závislosti priepustnosti na veľkosti dlhého rámca s minimálnou dĺžkou fragmentu 64 bajtov, Obr.4.3 zobrazuje grafické porovnanie závislosti priepustnosti na veľkosti dlhého rámca s minimálnou dĺžkou fragmentu 128 bajtov a na Obr.4.4 je zobrazené grafické porovnanie závislosti priepustnosti na veľkosti dlhého rámcu s minimálnou dĺžkou fragmentu 256 bajtov.

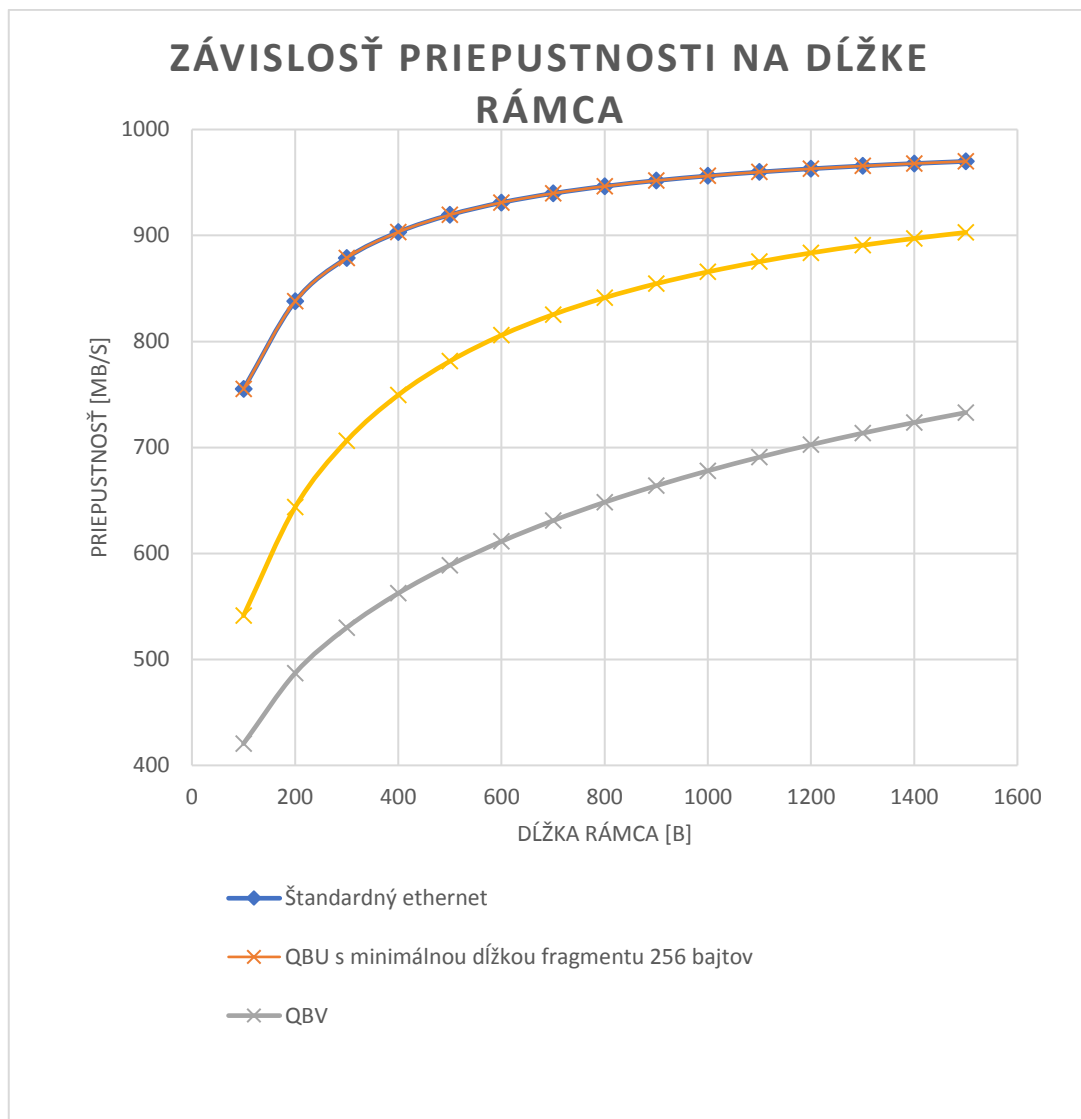
Na všetkých grafoch je možné vidieť, že priepustnosť štandardného ethernetu a 802.1QBV sa nemení v závislosti na minimálnej dĺžke fragmentu, keďže u nich neprebíha fragmentácia rámca. Štandard 802.1QBU sa priepustnosťou najviac približuje štandardnému ethernetu. Priepustnosť kombinácie 802.1QBU a 802.1QBV sa pohybuje medzi priepustnosťami štandardu 802.1QBV a štandardného ethernetu, priepustnosť klesá s rastúcou minimálnou dĺžkou fragmentu.



Obr. 4.2: Porovnanie závislostí priepustností pre štandardný ethernet QBV, QBU s minimálnou dĺžkou fragmentu 64 bajtov a kombinácie QBV s QBU s minimálnou dĺžkou fragment 64 bajtov.



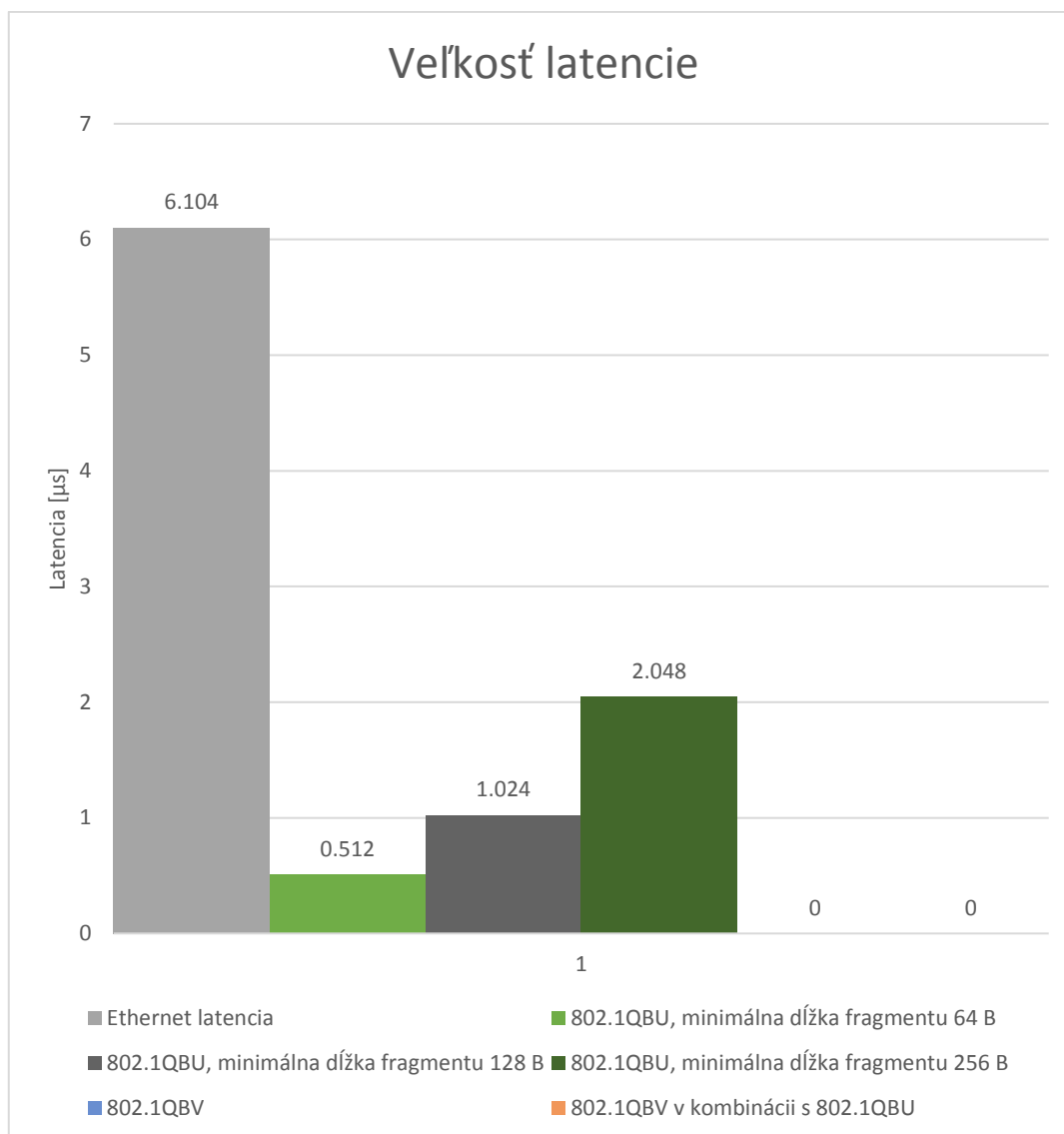
Obr. 4.3: Porovnanie závislostí priepustností pre štandardný ethernet QBV, QBU s minimálnou dĺžkou fragmentu 128 bajtov a kombinácie QBV s QBU s minimálnou dĺžkou fragment 128 bajtov.



Obr. 4.4: Porovnanie závislostí priepustností pre štandardný ethernet QBV, QBU s minimálnou dĺžkou fragmentu 256 bajtov a kombinácie QBV s QBU s minimálnou dĺžkou fragment 256 bajtov.

4.3 Porovnanie latencie

Na Obr.4.5 je uvedený graf, ktorý zobrazuje priemernú latenciu expresných rámcov (časovo kritické rámce) porovnávaných štandardov sieťovej komunikácie. Sieťové štandardy 802.1QBV a kombinácia štandardov 802.1QBU a 802.1QBV majú nulovú latenciu, pretože čas vysielania expresných rámcov je naplánovaný a počas ochranného pásma nezačína žiadna nová komunikácia. Pri komunikácii cez štandardný ethernet môže latencia časovo kritických rámcov dosahovať čas potrebný pre odoslanie rámca s maximálnou dĺžkou, preto je štandardný ethernet nevhodný pre aplikácie s časovo kritickou triedou sieťovej prevádzky. Latencia u štandardu 802.1QBU môže dosahovať od nulovej doby po čas potrebný na odoslanie rámca s dvojnásobnou dĺžkou ako je minimálna dĺžka fragmentu.



Obr. 4.5: Porovnanie priemernej veľkosti latencie pre expresné rámce.

5 IMPLEMENTÁCIA PODŠTANDARDU 802.1QBU

Pre implementáciu bola zvolená preempcia v izoláciu s dvomi triedami sieťovej prevádzky zo strany host-a, keďže podľa štandardu sa triedy sieťovej prevádzky podľa priority a konfigurácie priradujú do dvoch tried sieťovej prevádzky, expresnej a preempčnej, na vyššej podvrstve. Preempcia v izolácii bola zvolená z dôvodu jednoduchšej implementácie a nie je pri nej potrebné implementovať iný rozširujúci podštandard zo skupiny 802.1Q. Pre posielanie paketov medzi IP v FPGA sa používajú rozhrania ako Avalon ST a AXI. Pre vyššiu flexibilitu a pomocné signály, ktoré je možné zahrnúť do rozhrania Avalon ST sa ako rozhranie na vstupoch a výstupoch zvolilo ako prenosové rozhranie Avalon ST.

Navrhnutý dizajn pracuje na frekvencii 125 MHz. Skladá sa z vysielacej a prijímacej časti a dvoch komponentov CRC_remove.

Vysielacia časť zabaľuje dátové pakety z vyššej vrstvy do formátovaných rámcov na základe triedy sieťovej prevádzky. V prípade fragmentácie sú fragmenty formátované na základe ich poradia. Toto formátovanie je definované v štandarde 802.3BR, je ho možné vidieť v tabuľke 3-1.

Na konci každého rámcu sa pripája štvôr bajtové pole FCS, kvôli overeniu správnosti dát na strane príjemcu. FCS sa počíta za pomoci CRC algoritmu. Pri fragmentácii sa používa FCS, ktoré vychádza z inej konštanty ako štandardné FCS. To z dôvodu aby príjemca vedel, že to nie je celý rámec. Toto FCS je tiež definované v štandarde 802.3BR.

Fragmentované môžu byť jedine preempčné rámce. K fragmentácii dochádza, keď je expresný rámec pripravený na odvysielanie a preempčný rámec sa stále vysielal. Aby mohlo dôjsť k fragmentácii, musia byť dodržané tieto dve podmienky:

- Počet odvysielaných bajtov z preempčného rámcu musí byť rovný alebo väčší ako je minimálna dĺžka fragmentu.
- Počet neodvysielaných bajtov z preempčného rámcu musí byť rovný alebo väčší ako je minimálna dĺžka fragmentu.

Minimálna dĺžka fragmentu je 64 bajtov z toho posledné 4 bajty sú FCS. Takže minimálna dĺžka rámcu, ktorý môže byť fragmentovaný je 124 bajtov, 120 ak sa do dĺžky nezapočíta FCS.

Dátová časť rámcu musí mať dĺžku od 60 do 1514 bajtov. Pre dátové pakety, ktoré majú dátovú časť kratšiu ako je minimálna dĺžka, sa vykonáva takzvaný padding. Pre dátové pakety, ktoré majú dátovú časť dlhšiu ako je maximálna dĺžka, sa vykonáva ich skracovanie.

Medzi vysielaním jednotlivých rámcov sa po dobu 12 hodinových cyklov nezačína žiadne vysielanie.

Prijímacia časť kontroluje formát rámcov a podľa toho určuje či sa jedná o preempčný alebo expresný rámec. Pridané formátovacie bajty zahadzuje.

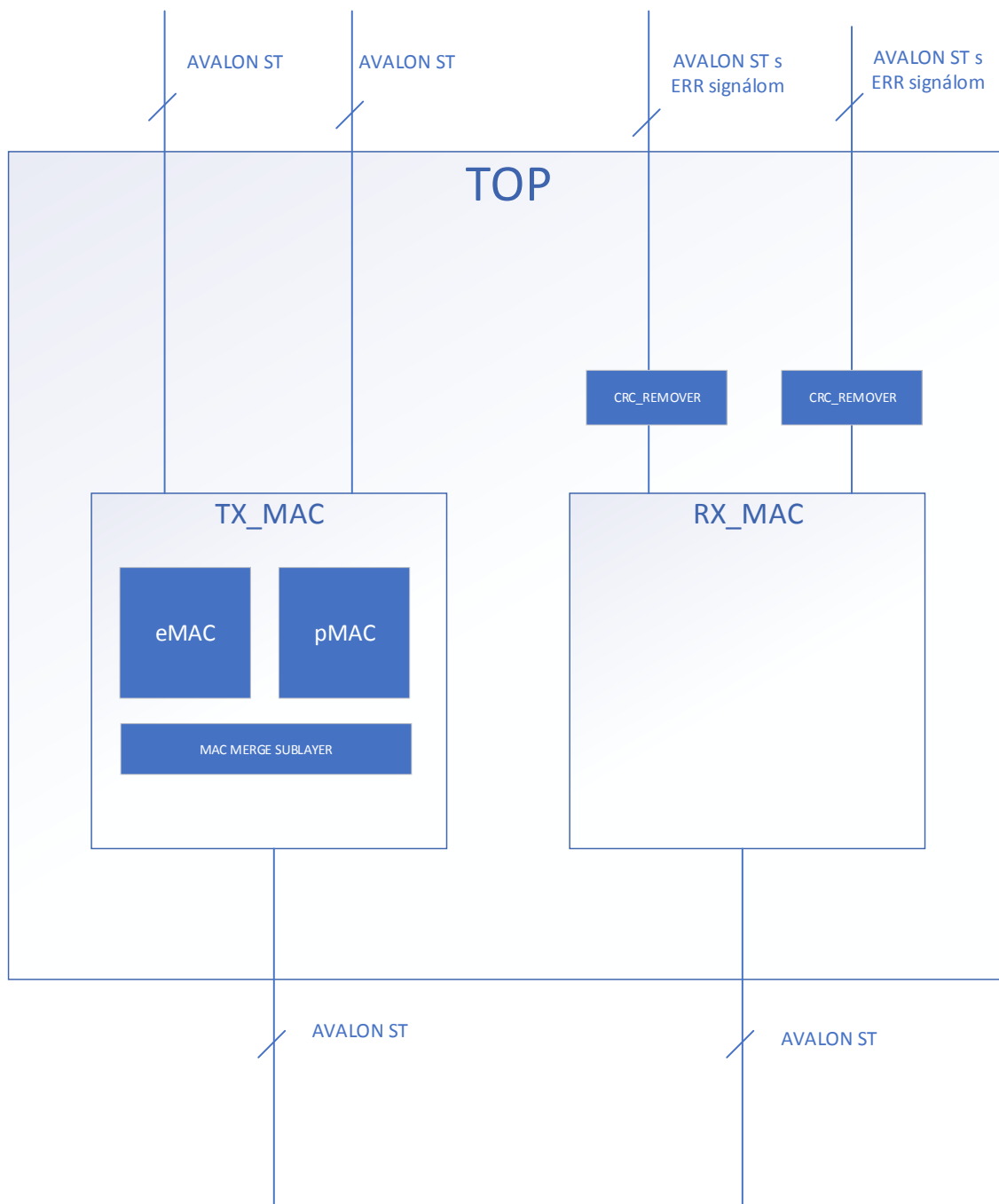
Správnosť prijatých dát kontroluje pomocou CRC algoritmu. Po skončení počítania sa výsledná hodnota porovná s konštantou. V prípade nerovnosti konštanty a výslednej

hodnoty, pošle prijímacia časť informáciu o chybných dátach vyššej vrstve.

Pri nesprávnej dĺžke rámca pošle prijímacia časť informáciu o chybných dátach vyššej vrstve, spolu s chybnými dátami.

Komponenty CRC_remover odstraňujú posledné štyri bajty z rámca (FCS).

Zapojenie dizajnu s použitými rozhraniami je zobrazené na Obr. 5.1.



Obr. 5.1: Top modul implementovaného dizajnu, s prepojením vnútorných komponentov.

5.1 RX MAC

RX MAC priíma rámce a na základe ich formátu určí do akej triedy sieťovej prevádzky spadajú, či sú chybné alebo fragmentované.

- Expresné rámce sú formátované tak, že na začiatku každého rámca je sedem bajtové pole, preambula vyplnená bajtmi s hodnotou 0x55 za ním nasleduje jeden bajt, SFD s hodnotou 0xD5. Nasledujúce bajty sú už dátová časť.
- Preempčné rámce a iniciálne fragmenty sú formátované na začiatku každého rámca, kde sa nachádza sedem bajtové pole, preambula vyplnená bajtmi s hodnotou 0x55 za ním nasleduje jeden bajt, SMDi, ktorá nadobúda štyri hodnoty: 0xE6, 0x4C, 0x7F a 0xB3, uvedené hodnoty sa v tomto poradí opakujú. Nasledujúce bajty sú dátová časť.
- Neiniciálne fragmenty sú formátované tak, že na začiatku každého rámca je šesť bajtové pole, preambula vyplnená bajtmi s hodnotou 0x55 za ním nasleduje jeden bajt, SMDc, ktorá nadobúda štyri hodnoty: 0x61, 0x52, 0x9E a 0xAD. Za SMDc nasleduje bajt, číslo fragmentu, ktoré nadobúda tieto hodnoty: 0xE6, 0x4C, 0x7F a 0xB3. Hodnoty pre SMDc a číslo fragmentu sa v tomto poradí opakujú a nasledujúce bajty sú už dátová časť.
- Všetky rámce s odlišným formátovaním sa vyhodnotia ako nesprávne a následne sa ich dátová časť zahodí.

Dátové rámce, ktoré sú kratšie ako 60 bajtov vyhodnotí prijímacia časť ako chybné. To isté urobí s rámcami, ktorých dátová časť je dlhšia 1514 bajtov. Uvedené platí aj pre rámce, ktoré boli fragmentované na kratšie fragmenty, čiže súčet dĺžky všetkých fragmentov pre jeden rámec musí byť menší ako 1514 bajtov.

Výsledok z CRC algoritmov sa musí rovnať konštante 0xDEBB20E3. V opačnom prípade ide o rámec s chybným dátovým poľom a prijímacia časť odošle informáciu vyššej vrstve, že ide o chybné dáta. Iniciálna hodnota pre CRC algoritmus je v prípade celých rámcov 0xFFFFFFFF, pri fragmentácii je iniciálna hodnota 0xFFFF0000 okrem koncového fragmentu, kde sa používa kontrola celého rámca.

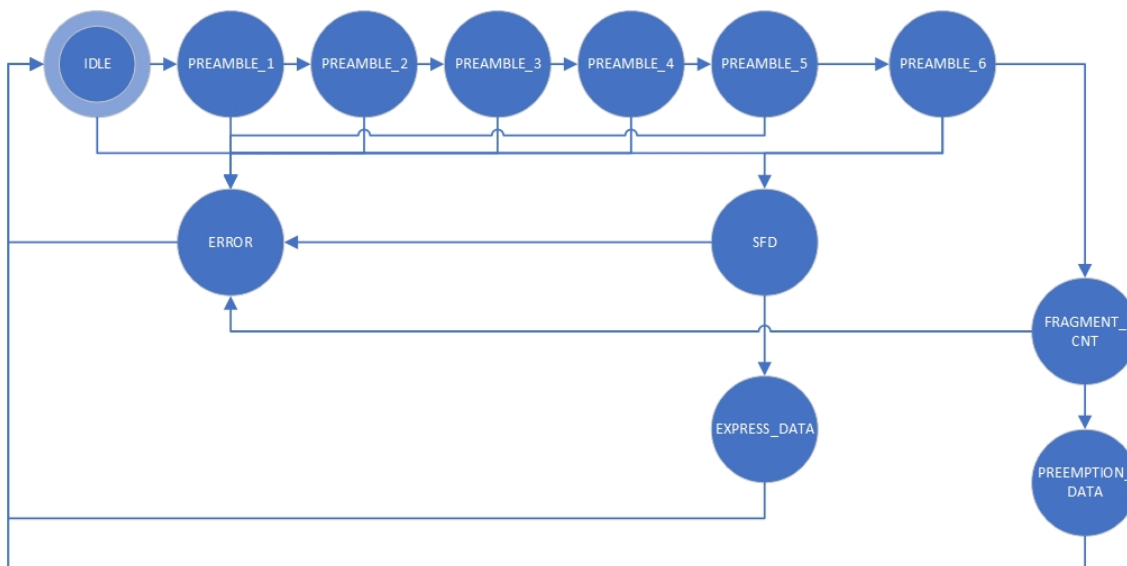
Prechody medzi stavmi sú vykreslené na Obr. 5.2. V tabuľke 5-1 sú vypísané výstupy z RX MAC pri danom stave. V tabuľke 5-2 sú popísané prechody medzi stavmi a podmienky, ktoré musia byť dodržané.

Tabuľka 5-1: Tabuľka prechodov medzi stavmi pre automat v TX MAC.

Stav	nasledujúci stav	podmienka
IDLE	PREAMBLE_1	sop=1 and data=0x55
PREAMBLE_1	PREAMBLE_2	data=0x55
	ERR_DATA	data≠0x55
PREAMBLE_2	PREAMBLE_3	data=0x55
	ERR_DATA	data≠0x55
PREAMBLE_3	PREAMBLE_4	data=0x55
	ERR_DATA	data≠0x55
PREAMBLE_4	PREAMBLE_5	data=0x55
	ERR_DATA	data≠0x55
PREAMBLE_5	PREAMBLE_5	data=0x55
	ERR_DATA	data≠0x55
PREAMBLE_6	PREAMBLE_6	data=0x55
	ERR_DATA	data≠0x55 and data≠SMDc
	FRG_CNT	data=SMDc
SFD	EXPRESS_DATA	data=SFD
	ERR_DATA	data≠SFD and data≠SMDi
	PREPMTION_DATA	data=SMDi
EXPRESS_DATA	IDLE	eop=1
	ERR_DATA	e_size > MAX_FRAME_DATA_BYTES + CRC32_BYTES
PREPMTION_DATA	IDLE	eop=1
	ERR_DATA	p_size > MAX_FRAME_DATA_BYTES + CRC32_BYTES
FRG_CNT	PREPMTION_DATA	data=next_frg
	ERR_DATA	data≠next_frg
ERR_DATA	IDLE	eop=0

Tabuľka 5-2: Tabuľka výstupov pre stavový automat v RX MAC.

Stav	Preempčná prevádzka				Expresná prevádzka			
	data	sop	eop	valid	data	sop	eop	valid
IDLE	0x00	0	0	0	0x00	0	0	0
PREAMBLE_1	0x00	0	0	0	0x00	0	0	0
PREAMBLE_2	0x00	0	0	0	0x00	0	0	0
PREAMBLE_3	0x00	0	0	0	0x00	0	0	0
PREAMBLE_4	0x00	0	0	0	0x00	0	0	0
PREAMBLE_5	0x00	0	0	0	0x00	0	0	0
PREAMBLE_6	0x00	0	0	0	0x00	0	0	0
EXPRESS_DATA	0x00	0	0	0	data_i	sop_i	eop_i	valid_i
PREEMPTION_DATA	data_i	sop_i	eop_i	valid_i	0x00	0	0	0
FRG_CNT	0x00	0	0	0	0x00	0	0	0
ERR_DATA	0x00	0	0	0	0x00	0	0	0



Obr. 5.2: Diagram stavového automatu v RX MAC.

5.2 TX MAC

Vysielacia časť prijíma dáta z vyššej vrstvy a podľa triedy sieťovej prevádzky ich formátuje:

Pred celé rámce a iníciaľne fragmenty sa na začiatok pridáva sedem bajtové pole, každý jeden bajt má hodnotu 0x55. Následne sa pridá jeden bajt pre expresnú triedu SFD s hodnotou 0xD5 a pre preempčnú triedu SMDi s hodnotami 0xE6, 0x4C, 0x7F a 0xB3. Tieto hodnoty sa opakujú v danom poradí, po SMDi alebo SFD nasleduje dátová časť.

Pred neiniciálne fragmenty sa vkladá šesť bajtové pole, všetky bajty majú hodnotu 0x55. Následne je pridaný bajt SMDc s hodnotami 0x61, 0x52, 0x9E a 0xAD, za SMDc

nasleduje bajt. Číslo fragmentu, ktoré nadobúda hodnoty: 0xE6, 0x4C, 0x7F a 0xB3, pričom hodnoty pre SMDc a číslo fragmentu sa v tomto poradí opakujú.

Rámce môžu byť odvysielané, jedine ak ich dátová časť má dĺžku aspoň 60 bajtov a ich dĺžka je kratšia alebo rovná 1514 bajtov. Rámce, ktoré majú dátovú časť kratšiu ako 60 bajtov sa za pomoci bajtov s hodnotou 0x00, rozširujú ich dátovú časť na minimálnu dĺžku. Dĺžka rámcu sa počíta pomocou čítača, ktorý sa inkrementuje, pri každom bajte od začiatku dát pri logickej jednotke signálu sop (start of packet) až po jeho koniec pri logickej jednotke signálu eop (end of packet).

Začiatok vysielania expresnej triedy sieťovej prevádzky má vždy vyššiu prioritu ako začiatok vysielania preempčnej triedy sieťovej prevádzky, alebo jej pokračovanie vo vysielaní. Expresná trieda má vyššiu prioritu než práve vysielajúci preempčný rámec.

K preempcii, prerušeniu vysielania rámcu, môže dôjsť jedine na preempčnej triede sieťovej prevádzky. Aby mohlo dôjsť k preempcii, musia byť splnené všetky nasledujúce podmienky:

- Počet odvysielaných bajtov je väčší ako je minimálna dĺžka fragmentu.
- Počet bajtov, ktoré ostávajú na odvysielanie, musí byť väčší ako je minimálna dĺžka fragmentu.
- Práve sa vysielal preempčný rámec.
- Expresný rámec je pripravený na vysielanie.

Na koniec každého rámcu alebo fragmentu z rámcu sa pridáva štvôrbajtové pole FCS, táto 32 bitová hodnota je výsledkom z CRC algoritmu, ktorým sa počíta z každého bajtu dátovej časti rámcu. Rámce a posledný fragment používajú ako iniciálnu hodnotu 0xFFFFFFFF, ostatné fragmenty používajú ako iniciálnu hodnotu 0xFFFF0000.

Medzi jednotlivými rámcami a fragmentami rámcov musí byť dodržaná medzera, ktorá musí mať dĺžku 12 bajtov, takzvaný interframe gap, ktorý je definovaný v štandarde 802.3 ako štandardný ethernet. Keďže sa používa osem bitová dátová šírka vysielaných dát, tak po odvysielaní rámcu alebo fragmentu sa nesmie začať žiadne nové vysielanie po dobu 12 hodinových cyklov.

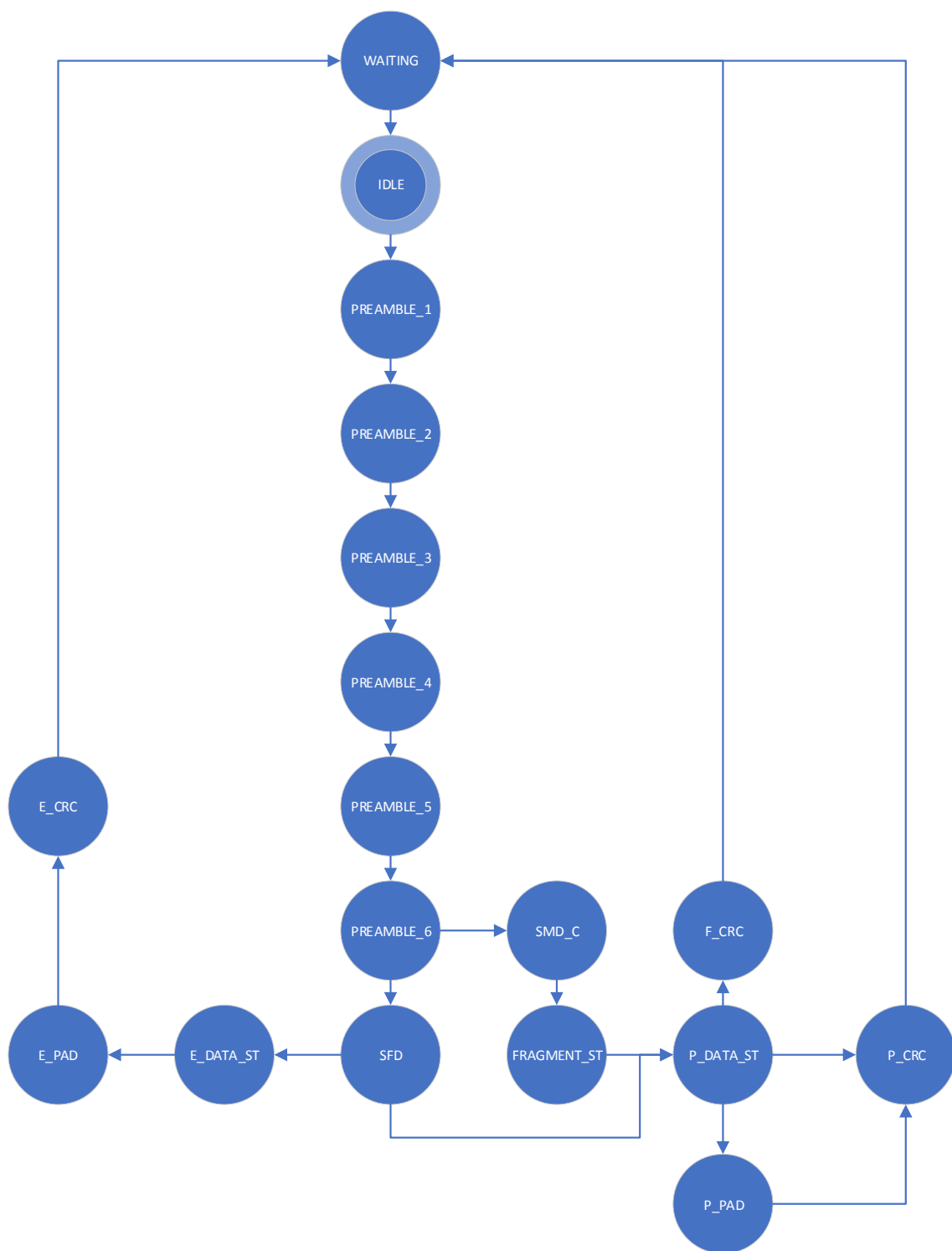
Prechody medzi stavmi sú vykreslené na Obr.7.3. V tabuľke 5-4 sú vypísané výstupy z RX MAC-u pri danom stave. V tabuľke 5-3 sú popísané prechody medzi stavmi a podmienky, ktoré musia byť dodržané.

Tabuľka 5-3: Tabuľka prechodov medzi stavmi pre automat v TX MAC.

Stav	Nasledujúci stav	podmienka
idle	preamble_1	e_sop or e_long or p_sop or p_long or fragment_cnt > 0
preamble_1	preamble_2	
preamble_2	preamble_3	
preamble_3	preamble_4	
preamble_4	preamble_5	
preamble_5	preamble_6	transmission = 11
	smd_c	transmission ≠ 11
preamble_6	sfd	
sfd	e_data_st	transmission = 01
	p_data_st	transmission = 10
smd_c	fragment_st	
fragment_st	p_data_st	
e_data_st	e_pad	e_eop and e_data_count < 59
	e_crc	e_eop and e_data_count > 59
p_data_st	p_pad	p_eop = 1 and p_data_count < 59
	f_crc	e_sop and data_count > 59 and (p_data_count + 59) pkt_size
	p_crc	p_eop = 1 and p_data_count > 63 or p_data_count = MAX_FRAME_DATA_BYTES
e_pad	e_crc	e_data_count > 59
p_pad	p_crc	p_data_count > 59
e_crc	waiting	crc_count = 3
f_crc	waiting	crc_count = 3
p_crc	waiting	crc_count = 3
waiting	idle	wait_count = 12

Tabuľka 5-4: Tabuľka výstupov pre stavový automat v TX MAC.

Stav	data			valid	eop	sop
	transmission=10	transmission=01	transmission=11			
idle	0x55	0x55	0x55	1	0	1
preamble_1	0x55	0x55	0x55	1	0	0
preamble_2	0x55	0x55	0x55	1	0	0
preamble_3	0x55	0x55	0x55	1	0	0
preamble_4	0x55	0x55	0x55	1	0	0
preamble_5	0x55	0x55	0x55	1	0	0
preamble_6	0x55	0x55	0x00	1	0	0
sfd	0xD5	SMD-Ix	0x00	1	0	0
smd_c	0x00	0x00	SMD-Cx	1	0	0
fragment_st	0x00	0x00	FRG_cnt	1	0	0
e_data_st	e_data	0x00	0x00	1	0	0
p_data_st	0x00	p_data	p_data	1	0	0
e_pad	0x00	0x00	0x00	1	0	0
p_pad	0x00	0x00	0x00	1	0	0
e_crc	exp_crc	0x00	0x00	1	if(crc=3) 1	0
f_crc	0x00	frag_crc	frag_crc	1	if(crc=3) 1	0
p_crc	0x00	preemp_crc	preemp_crc	1	if(crc=3) 1	0
waiting	0x00	0x00	0x00	1	0	0



Obr. 5.3: Diagram stavového automatu v TX MAC.

5.3 CRC_REMOVER

Uvedený komponent odstraňuje posledné štyri bajty z dát obdržaných na prijímacej strane a preposiela všetky signály zahrnuté v rozhraní Avalon ST smerom k hostovi. Signál eop z Avalon ST rozhrania príde k hostovi o štyri takty skôr ako by mal, čo je spôsobené tým, že posledné štyri bajty boli odstránené a tým skôr sa uskutočnil koniec vysielaných dát. Podobne je to s potvrdzovacím signálom valid, jeho trvanie sa skrúti tiež o štyri takty z tých istých dôvodov.

5.4 Avalon Streaming

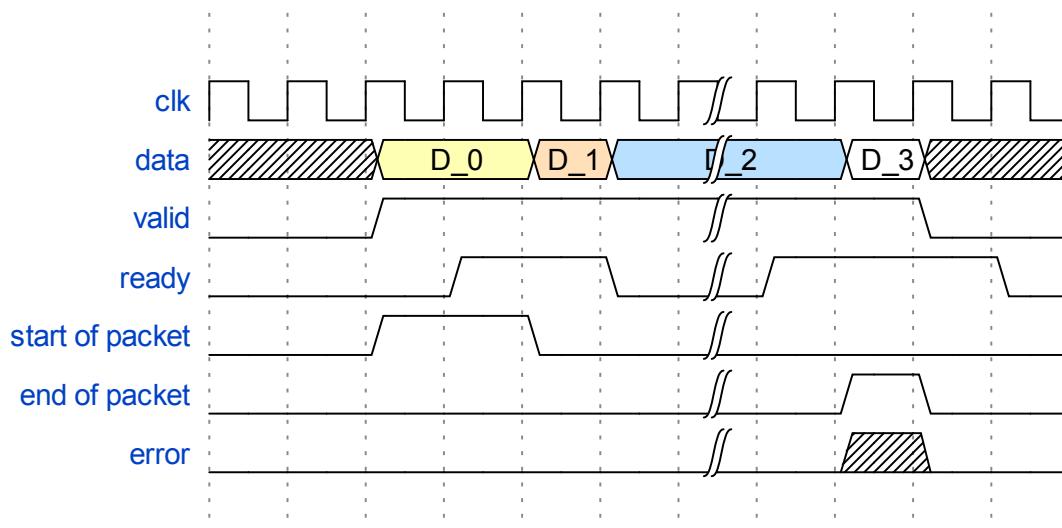
Rozhranie Avalon ST je podporované v FPGA od Intel FPGA. Komunikácia na tomto rozhraní prebieha medzi vysielacou stranou (source) a prijímacou stranou (sink). Základná verzia má len tri signály:

- clk je hodinový signál, podľa ktorého sa riadi Avalon ST rozhranie.
- valid je signál zo source do sink, týmto signálom potvrdzuje source, že sa jedná o platné dáta. Dáta bez valid signálu sink neprijíma.
- data je signál zo source do sink, tento signál nesie informácie. Môže mať dátovú šírku od 1 do 4096 bitov. Pre dizajn bola zvolená 8 bitová šírka.

Pre Avalon ST existujú ďalšie rozširujúce signály. Z týchto rozširujúcich signálov boli použité štyri:

- ready je signál zo sink do source. Tento signál dáva source informáciu, či je sink pripravený prijímať dáta. Pokiaľ je ready nastavený na logickú nulu, potom source neposiela ďalšie dáta do sink.
- error je signál zo source do sink. Tento signál je bitová maska, ktorá informuje o chybách pri prenose údajov. Každý bit informuje o inej chybe, ktorá nastala pri prenose, bitová šírka môže byť od 1 do 256 bitov. V dizajne sa tento signál nachádza iba na výstupe z RX strany. Šírka pre tento signál bola zvolená na 1 bit.
- endofpacket je signál zo source do sink. Tento signál sa nastaví do logickej jednotky na začiatku vysielania dátového paketu, inak je v logickej nule.
- startofpacket je signál zo source do sink. Tento signál sa nastaví do logickej jednotky na konci vysielania dátového paketu, inak je v logickej nule.

Pre lepšie pochopenie fungovania tohto rozhrania je uvedený príklad na Obr.5.4. Kde je možné vidieť, že sink si vie regulovať prichádzajúce dáta informovaním source cez signál ready, vďaka tomu sa môže prerušiť vysielanie dátového paketu, kým sink nedovolí pre source jeho pokračovanie.



Obr. 5.4: Príklad fungovania komunikačného rozhrania Avalon ST

5.5 Syntéza

Pre správne fungovanie dizajnu je potrebné, aby minimálna frekvencia dizajnu bola vyššia ako 125 MHz, jej znížením by sa musela znížiť prenosová rýchlosť, tá bola stanovená na 1 Gb/s. Pre syntézu s týmto dizajnom vyšlo, že minimálna frekvencia tohto dizajnu pre FPGA z rodiny STRATIX IV, konkrétne zariadenie EP4SGX70HF35C2 pre pracovnú teplotu 85 °C je maximálna frekvencia 161,55 MHz a pre pracovnú teplotu 0 °C je maximálna frekvencia 168,35 MHz.

Ďalej zo syntézy vyšlo, že pre zvolené FPGA dizajn spotrebuje 1400 z 58080 kombinačný ALUT tabuliek, čo je využitie na 2 % a ešte spotrebuje 973 z 58080 logických registrov. Výsledky a syntézy časovej analýzy sú zhrnuté v tabuľke 7-5.

Tabuľka 5-5: Prehľad výsledkov syntézy.

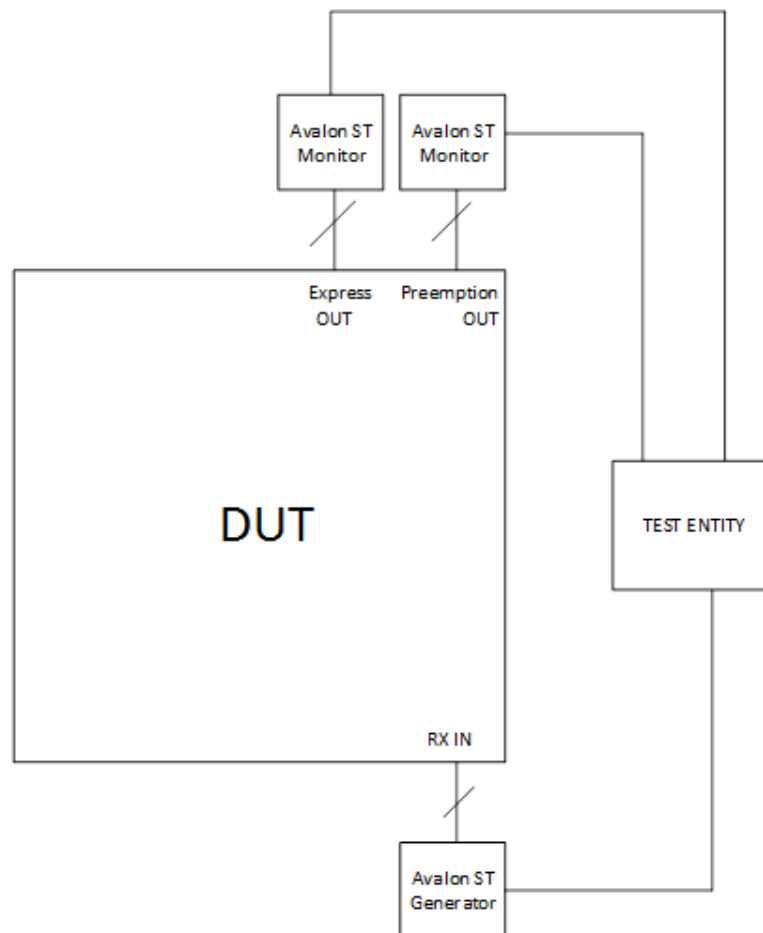
Maximálna pracovná frekvencia pri 85 °C	161,55 MHz
Maximálna pracovná frekvencia pri 0 °C	168,35 MHz
ALUT	1400
Logické registre	973

6 SIMULÁCIE

Pre odsimulovanie a otestovanie dizajnu boli použité dve testovacie prostredia. Prvý *tb_top_rx* testuje a simuluje iba RX stranu finálneho dizajnu. Naopak testovacie prostredie *tb_top_final* testuje celý dizajn, kde sa RX strana používa pre kontrolu formátu odoslaných rámcov z TX strany.

6.1 Testovacie prostredie *tb_top_rx*

V tomto testovacom prostredí sú všetky výstupy z TX strany zadané ako open, čiže nie sú nikde pripojené, na vstupy TX strany sú pripojené logické jednotky. Na vstup RX strany je pripojený generátor rámcov s rozhraním Avalon ST, to isté ako je použité v celom dizajne až na výstup z RX strany. Na výstup z RX strany sú pripojené dva monitory, ktoré zachytávajú a porovnávajú rámce. Tieto monitory majú tiež Avalon ST rozhranie, ale rozšírené o Error signál, presne tak ako na výstupe RX strany.



Obr. 6.1: Bloková schéma testovacieho prostredia *tb_top_rx*.

V tomto testovacom prostredí sa vykonávajú štyri nezávislé test procedúry, ktoré sú popísané v nasledujúcich podkapitolách. Tieto test procedúry sa snažia čo najviac pokryť všetky funkcie dizajnu RX strany a overiť jej správnu funkčnosť. Na Obr. 6.1 je znázornené zapojenie testovaného dizajnu a verifikačných BFM.

6.1.1 Test procedúra tp_preamble

Uvedená test procedúra sa využíva na overenie funkcie dizajnu, že rámce s nesprávnou preambulou, s nesprávnym SFD, SMD alebo nesprávnym poradím fragmentu sú vyhodnotené ako chybové a sú zahodené. Pre porovnanie správnosti boli použité rámce s korektnou preambulou a korektným SFD, SMD a správnym poradím fragmentov.

Testovanie prebiehalo následne, najprv boli posielané rámce s chybnou časťou preamble, od chyby preamble v prvom bajte až po chybu preamble v poslednom bajte, siedmom. Následne bola vykonaná chyba v ôsmom bajte rámca, bola nastavená hodnota, ktorá nie je zhodná s hodnotou SFD a zároveň nie je zhodná s hodnotami SMD-Cx. Po odoslaní týchto rámcov sa odoslal referenčný rámec s korektnými bajtmi v preamble a so správnou hodnotou SFD, dĺžka dátovej časti rámca bola 64 bajtov. Dátová časť rámca bola vygenerovaná pre ľahšiu kontrolu inkrementáciou iniciálnej hodnoty, v tomto prípade 0x11. Na záver dátovej časti bolo pripojené štvôr bajtové pole s korektným FCS, ktoré sa vypočítalo pomocou funkcie vygenerovanej zo zdroja na internete. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-1.

Tabuľka 6-1: Inštrukcie v test procedúre tp_preamble.

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	Generátor: 0x11555555555555D5 + 64 B + FCS	-
13620	Generátor: 0x55115555555555D5 + 64 B + FCS	-
16860	Generátor: 0x55551155555555D5 + 64 B + FCS	-
20100	Generátor: 0x55555511555555D5 + 64 B + FCS	-
23340	Generátor: 0x55555555115555D5 + 64 B + FCS	-
26580	Generátor: 0x55555555551155D5 + 64 B + FCS	-
29820	Generátor: 0x55555555555511D5 + 64 B + FCS	-
33060	Generátor: 0x5555555555555511 + 64 B + FCS	-
36300	Generátor: 0x55555555555555D5 + 64 B + FCS	Dátová časť (64 B)
39660	Ukončenie test procedúry	-

Z výsledkov simulácie vyšlo, že žiadny rámec alebo fragment rámca s chybnou preambulou, SFD, SMD alebo poradím fragmentu neprejde cez RX stranu ďalej do hosta. Rámce so správnymi inicializačnými poľami sa začnú spracovávať a v prípade ich správnej veľkosti a správnej výslednej hodnoty po cyklickej redunďnej kontrole sa ďalej posielajú ako validné..

6.1.2 Test procedúra tp_express

Cieľom test procedúry je overiť všetky možnosti na RX strane, ktoré môžu vzniknúť pri prijímaní expresných rámcov. RX strana by mala zareagovať na expresné rámce s malou dĺžkou, malou dĺžkou a chybným FCS, veľkou dĺžkou, veľkou dĺžkou a chybným FCS a dĺžkou s povoleným rozsahom, alebo chybným FCS tak, že ich vyhodnotí ako chybné a na koniec rámcu nastaví error signál do logickej úrovne jedna. V prípade dlhých rámcov sa po presiahnutí povolenej dĺžky nastaví error signál tiež do jednotky a prestanú sa dáta z rámcu ďalej posielat'.

Testovanie prebiehalo následne: pre všetky posielané rámce bola poslaná korektná preambula a SFD, dátová časť generovala inkrementáciu iniciálnej hodnoty, ktorá bola vo všetkých prípadoch nastavená na hodnotu 0x11. V prípade, že na konci rámcu malo byť správne FCS v závere bolo pripojené štvôr bajtové pole, ktoré bolo spočítané pomocou funkcie vygenerovanej na internete. Najskôr boli odoslané dva rámce s malou dĺžkou 20 bajtov, jeden s pridaným poľom so správnym FCS, čiže tento rámec mal celkovú dĺžku 24 bajtov plus preambula a SFD. Dáta očakávané na výstupe boli nastavené v monitore s dĺžkou 16 bajtov pre rámec bez FCS a s dĺžkou 20 bajtov pre rámec s platným FCS. Následne boli odoslané dva rámce s dĺžkou 200 bajtov, jeden s poľom obsahujúcim FCS a druhý bez tohto poľa. Pre kontrolu rámcu s platným FCS boli nastavené dáta s dĺžkou 200 bajtov a pre rámec bez FCS poľa bolo nastavenie dát s dĺžkou 196 bajtov. Nakoniec boli odoslané dva rámce s dĺžkou 2000 bajtov, pričom maximálna dĺžka dátovej časti je 1520 bajtov. K prvému rámcu bolo pridané pole s platným FCS a k druhému sa nič nepridalo. Pre odkontrolovanie oboch rámcov boli nastavené dáta s dĺžkou 1520 bajtov, keďže RX strana nič nerobí s dátami ak počet bajtov presiahne maximálnu dĺžku rámcu. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-2.

Tabuľka 6-2: Inštrukcie v test procedúre tp_express

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	expresný rámec s celkovou dĺžkou 20 B a s platným CRC	20 B a error signál = '1'
11820	expresný rámec s celkovou dĺžkou 20 B a bez platného CRC	16 B a error signál = '1'
13100	expresný rámec s celkovou dĺžkou 200 B a s platným CRC	200 B
21740	expresný rámec s celkovou dĺžkou 200 B a bez platného CRC	196 B a error signál = '1'
30220	expresný rámec s celkovou dĺžkou 2000 B a s platným CRC	1516 B a error signál = '1'
110900	expresný rámec s celkovou dĺžkou 2000 B a bez platného CRC	1520 B a error signál = '1'
191420	Ukončenie test procedúry	-

Z výsledkov simulácie je zjavné, že rámce s neplatnou dĺžkou a neplatným FCS sú vyhodnotené ako chybové a posielajú sa ďalej s chybovou hláškou pre informovanie vyššej vrstvy, to isté platí aj pre rámce s nesprávnou dĺžkou.

6.1.3 Test procedúra tp_no_preemption

Cieľom tejto test procedúry bolo overiť všetky možnosti na RX strane, ktoré môžu vzniknúť pri prijímaní preempčných rámcov. RX strana by mala zareagovať na preempčné rámce s malou dĺžkou, malou dĺžkou a chybným FCS, veľkou dĺžkou, veľkou dĺžkou a chybným FCS a dĺžkou s povoleným rozsahom ale chybným FCS, tak že ich vyhodnotí ako chybné a na koniec rámca nastaví error signál do logickej úrovne jedna, v prípade dlhých rámcov sa po presiahnutí povolenej dĺžky nastaví error signál tiež do jednotky a dáta z rámca sa prestanú ďalej posielat'. Ťažšie to je s SMD, pretože sa mení pri každom rámci a celkovo sa striedajú štyri hodnoty, ktoré sa cyklija po sebe.

Pre správne testovanie bolo potrebné pri test procedúre poslať päť rámcov po sebe pre overenie cyklenia hodnôt SMD z každej testovanej kategórii. Pre všetky rámce bola generovaná preambula spolu so správnym SMD. Dátová časť pre všetky rámce je generovaná inkrementáciou z iniciálnej hodnoty.

- Na začiatku bolo odoslaných päť rámcov z dlhých 200 bajtov, preto na výstupe monitor očakáva päť rámcov po 200 bajtoch.
- V ďalšom teste bolo odoslaných päť rámcov po 200 bajtoch, ale na konci nebolo pridané FCS, čo spôsobí odstrihnutie posledných štyroch bajtov, takže na výstupe monitor očakáva päť rámcov s dĺžkou 196 bajtov. Pre správne dodržanie SMD prvý rámec sa posielal nie s nulovou hodnotou SMD, ale už s prvou.
- Následne sa odošle päť rámcov dlhých 2000 bajtov bez FCS a potom päť bajtov s platným FCS, pri všetkých týchto rámcoch sedí SMD. Pre testovanie oboch týchto sérií sa na monitor nastaví rámce s 1520 bajtov, zvyšok rámcu sa zahodí na porovnanie s odoslanými dátami.
- Nakoniec sa odošle päť rámcov s dĺžkou 20 bajtov a platným FCS a päť rámcov s dĺžkou 20 bajtov a neplatným FCS, pre všetky rámce sa vygeneruje správne SMD. Pre monitor na výstupe sa nastaví 20 bajtov na porovnanie s rámcami, ktoré obsahujú FCS. Pre rámce, ktoré neobsahujú FCS sa nastaví na monitor 16 bajtov.

Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-3.

Tabuľka 6-3: Inštrukcie v test procedúre tp_no_preemption

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	preempčný rámec s celkovou dĺžkou 200 B a s platným CRC a SMD s poradím 0	Dáta 200 B
19060	preempčný rámec s celkovou dĺžkou 200 B a s platným CRC a SMD s poradím 1	Dáta 200 B
27740	preempčný rámec s celkovou dĺžkou 200 B a s platným CRC a SMD s poradím 2	Dáta 200 B
36420	preempčný rámec s celkovou dĺžkou 200 B a s platným CRC a SMD s poradím 3	Dáta 200 B

45100	preempční rámec s celkovou délkou 200 B a s platným CRC a SMD s poradím 0	Dáta 200 B
53780	preempční rámec s celkovou délkou 200 B a bez platného CRC a SMD s poradím 1	Dáta 196 B a error signál = '1'
62300	preempční rámec s celkovou délkou 200 B a bez platného CRC a SMD s poradím 2	Dáta 196 B a error signál = '1'
70820	preempční rámec s celkovou délkou 200 B a bez platného CRC a SMD s poradím 3	Dáta 196 B a error signál = '1'
79340	preempční rámec s celkovou délkou 200 B a bez platného CRC a SMD s poradím 0	Dáta 196 B a error signál = '1'
87860	preempční rámec s celkovou délkou 200 B a bez platného CRC a SMD s poradím 1	Dáta 196 B a error signál = '1'
96380	preempční rámec s celkovou délkou 2000 B a s platným CRC a SMD s poradím 2	Dáta 1520 B a error signál = '1'
177060	preempční rámec s celkovou délkou 2000 B a s platným CRC a SMD s poradím 3	Dáta 1520 B a error signál = '1'
257740	preempční rámec s celkovou délkou 2000 B a s platným CRC a SMD s poradím 0	Dáta 1520 B a error signál = '1'
338420	preempční rámec s celkovou délkou 2000 B a s platným CRC a SMD s poradím 1	Dáta 1520 B a error signál = '1'
419100	preempční rámec s celkovou délkou 2000 B a s platným CRC a SMD s poradím 2	Dáta 1520 B a error signál = '1'
499780	preempční rámec s celkovou délkou 2000 B a bez platného CRC a SMD s poradím 3	Dáta 1520 B a error signál = '1'
580300	preempční rámec s celkovou délkou 2000 B a bez platného CRC a SMD s poradím 0	Dáta 1520 B a error signál = '1'
660820	preempční rámec s celkovou délkou 2000 B a bez platného CRC a SMD s poradím 1	Dáta 1520 B a error signál = '1'
741340	preempční rámec s celkovou délkou 2000 B a bez platného CRC a SMD s poradím 2	Dáta 1520 B a error signál = '1'
821860	preempční rámec s celkovou délkou 2000 B a bez platného CRC a SMD s poradím 3	Dáta 1520 B a error signál = '1'
902380	preempční rámec s celkovou délkou 20 B a s platným CRC a SMD s poradím 0	Dáta 20 B a error signál = '1'
903860	preempční rámec s celkovou délkou 20 B a s platným CRC a SMD s poradím 1	Dáta 20 B a error signál = '1'
905340	preempční rámec s celkovou délkou 20 B a s platným CRC a SMD s poradím 2	Dáta 20 B a error signál = '1'
906820	preempční rámec s celkovou délkou 20 B a s platným CRC a SMD s poradím 3	Dáta 20 B a error signál = '1'
908300	preempční rámec s celkovou délkou 20 B a s platným CRC a SMD s poradím 0	Dáta 20 B a error signál = '1'
909780	preempční rámec s celkovou délkou 20 B a bez platného CRC a SMD s poradím 1	Dáta 16 B a error signál = '1'
911100	preempční rámec s celkovou délkou 20 B a bez platného CRC a SMD s poradím 2	Dáta 16 B a error signál = '1'
912420	preempční rámec s celkovou délkou 20 B a bez platného CRC a SMD s poradím 3	Dáta 16 B a error signál = '1'

913740	preempčný rámec s celkovou dĺžkou 20 B a bez platného CRC a SMD s poradím 0	Dáta 16 B a error signál = '1'
915060	preempčný rámec s celkovou dĺžkou 20 B a bez platného CRC a SMD s poradím 1	Dáta 16 B a error signál = '1'
926380	Ukončenie test procedúry	-

Z výsledkov simulácie vyplýva, že rámce s neplatnou dĺžkou a neplatným FCS sú vyhodnotené ako chybové a posielajú sa ďalej s chybovou hláškou, pre informovanie vyššej vrstvy, to isté platí aj pre rámce s nesprávnou dĺžkou. Ďalej sa bolo overené, že cyklenie SMD je potrebné pre posielanie rámcov na RX stranu.

6.1.4 Test procedúra tp_preemption

Cieľom tejto test procedúry je overiť všetky možnosti na RX strane, ktoré môžu vzniknúť pri prijímaní fragmentov preempčných rámcov. RX strana by mala zareagovať na preempčné rámce s malou dĺžkou, malou dĺžkou a chybným FCS, veľkou dĺžkou, veľkou dĺžkou a chybným FCS a dĺžkou s povoleným rozsahom, ale chybným FCS tak, že ich vyhodnotí ako chybné a na koniec rámca nastaví error signál do logickej úrovni jedna. V prípade dlhých rámcov sa po presiahnutí povolenej dĺžky nastaví error signál tiež do jednotky a prestanú sa dáta z rámca ďalej posielat'. Náročnejšie to je s SMD, ktorý sa mení pri každom rámci, celkovo sa striedajú štyri hodnoty, ktoré sa cyklija po sebe.

Pri tejto test procedúre bolo na začiatku poslaných šesť rámcov s dĺžkou 700 bajtov, ktoré boli rozdelené na sedem fragmentov po 100 bajtoch s dodržaním správneho formátu a s korektnými poľami obsahujúcimi FCS. Následne bol odoslaný jeden rámec dlhý 700 bajtov, ktorý bol rozdelený na sedem fragmentov. Všetky fragmenty obsahujú chybné pole s FCS až na iniciálny fragment, ten obsahuje korektné pole s FCS. V ďalšom kroku bol opäť odoslaný rámec s dĺžkou 700 bajtov, rozdelený na sedem fragmentov po 100 bajtoch, všetky fragmenty obsahujú pole so správnym FCS, ale obsahujú nesprávne poradie fragmentu. Z uvedeného vyplýva, že na vyššiu vrstvu sa dostane len iniciálny fragment. Následne boli odoslané dva rámce s dĺžkou 10 bajtov, prvý obsahoval pole so správnym FCS a ten druhý nie. To znamená, že z prvého rámca by mali prejsť všetky dátové bajty s propagovaným error signálom a z druhého rámca by mali prejsť len prvých šesť bajtov, keďže posledné štyri bajty by mali byť odstrihnuté pre neprítomnosť FCS poľa. V závere testu boli odoslané dva 7000 bajtové rámce, ktoré nesplňajú rozmery štandardného rámca. Obidva boli rozdelené na fragmenty s dĺžkou 1000 bajtov. Prvý fragment z rámcov prešiel bez problémov, ale z ostatných fragmentov prešli dáta po 520 bajtov. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-4.

Tabuľka 6-4: Inštrukcie v test procedúre tp_preemption

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
380	preempčný rámec so 7 fragmentami po 100 B a s platným CRC a SMD s poradím 0	Dáta 6 * 100 B
45660	preempčný rámec so 7 fragmentami po 100 B a s platným CRC a SMD s poradím 1	Dáta 6 * 100 B

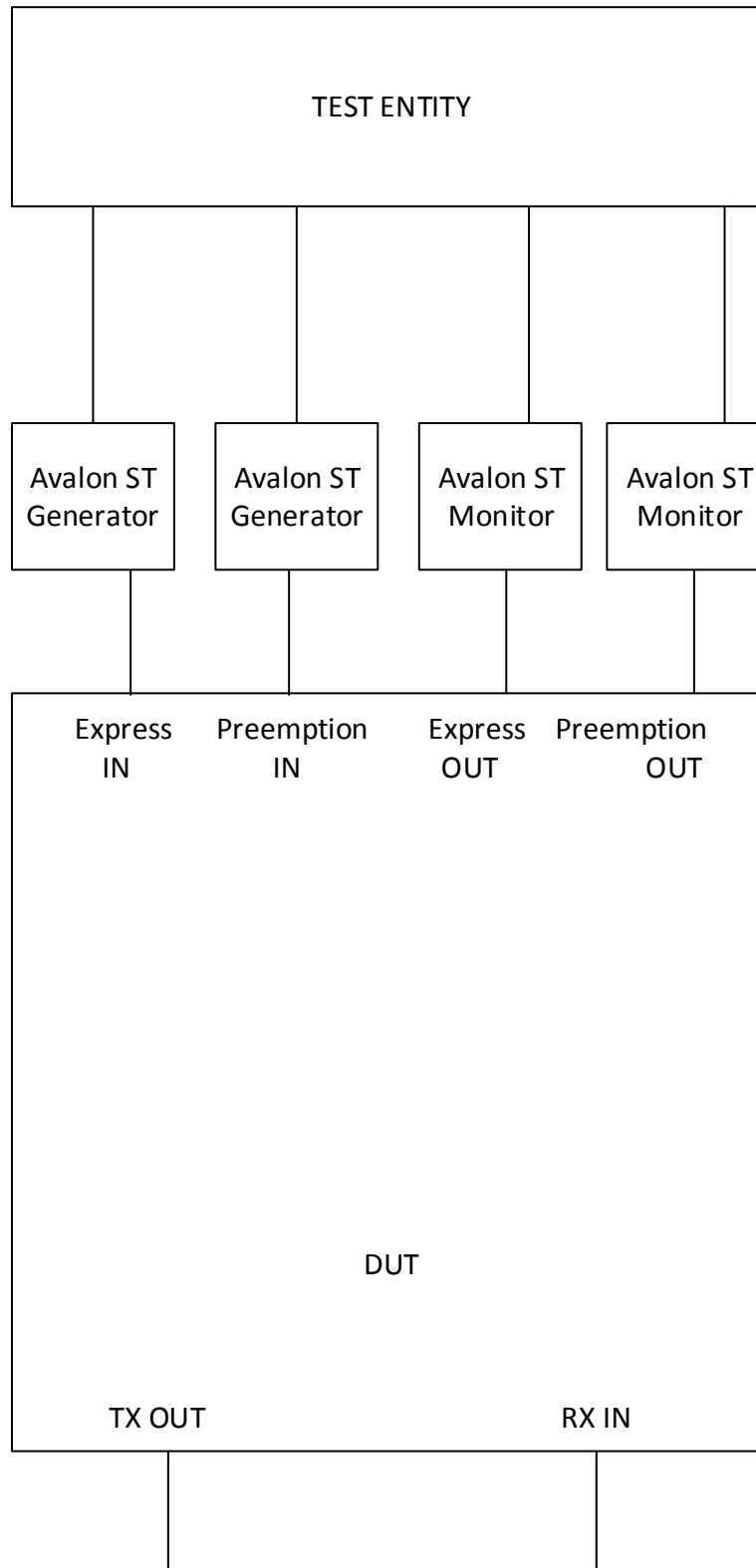
90940	preempčný rámec so 7 fragmentami po 100 B a s platným CRC a SMD s poradím 2	Dáta 6 * 100 B
136220	preempčný rámec so 7 fragmentami po 100 B a s platným CRC a SMD s poradím 3	Dáta 6 * 100 B
181500	preempčný rámec so 7 fragmentami po 100 B a s platným CRC a SMD s poradím 0	Dáta 6 * 100 B
226780	preempčný rámec so 7 fragmentami po 100 B a s platným CRC a SMD s poradím 1	Dáta 6 * 100 B
272060	preempčný rámec so 7 fragmentami po 100 B a bez platného CRC a SMD s poradím 2	Dáta 6*96 B s error signálom = '1' na konci každého fragmentu
307340	preempčný rámec so 6 fragmentami po 100 B a s platným CRC a neplatným poradím čísla fragmentu a SMD s poradím 3	Dáta 100 B, ďalšie fragmenty sa zahodia
342380	preempčný rámec so 6 fragmentami po 10 B a bez platného CRC a SMD s poradím 0	Dáta 6*6 B s error signálom = '1' na konci každého fragmentu
352460	preempčný rámec so 6 fragmentami po 10 B a s platným CRC a SMD s poradím 1	Dáta 6*10 B s error signálom = '1' na konci každého fragmentu
362540	preempčný rámec so 6 fragmentami po 1000 B a s platným CRC a SMD s poradím 2	Dáta 1000 B + 520 B, ostatné fragmenty sa zahodia a s error signálom = '1' na konci každého fragmentu
659820	Ukončenie test procedúry	

Pomocou test procedúry sa podarilo overiť fungovanie skladania fragmentov do celých rámcov na RX strane. Ku kompletnému poskladaniu rámcov dôjde, jedine v prípade, že bol dodržaný správny formát rámca. V inom prípade sa do vyššej vrstvy propaguje chybová informácia v tvare error signálu.

6.2 Testovacie prostredie tb_top_final

V tomto testovacom prostredí je výstup z TX strany napojený na vstup RX strany. Na obidva vstupy pre TX stranu sú napojené Avalon ST generátory. Na výstup z RX strany sú pripojené dva monitory, ktoré zachytávajú a porovnávajú rámce. Tieto monitory majú tiež Avalon ST rozhranie, ale rozšírené o Error signál presne tak, ako na výstupe RX strany.

V tomto testovacom prostredí sa vykonávajú tri nezávislé test procedúry, ktoré sú popísané v nasledujúcich podkapitolách. Tieto test procedúry sa snažia čo najviac pokryť všetky funkcie dizajnu TX strany a overiť jej správnu funkčnosť, keďže RX strana bola overená v testovacom prostredí tb_top_final a môže sa použiť ako otestovaný prijímač. Na Obr. 8.2 je znázornené zapojenie testovaného dizajnu a verifikačných BFM.



Obr. 6.2: Bloková schéma testovacieho prostredia `tb_top_final`.

6.2.1 Test procedúra tp_padding

Cieľom tejto test procedúry je overiť, či pri krátkom dátovom pakete sa vygeneruje TX strana rámce s minimálnou dĺžkou, kde zvyšné dátové pole bude vyplnené bajtmi s hodnotou 0x00.

Pre testovanie boli vygenerované rámce s dĺžkou 10 bajtov. Očakávané rámce na monitoroch boli nastavené na dĺžku 60 bajtov, ale len prvých 10 bajtov bolo generovaných z počiatočnej hodnoty 0x11 za pomoci inkrementácie. Zvyšné bajty v očakávanom rámci boli nastavené na hodnotu 0x00. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-5.

Tabuľka 6-5: Inštrukcie v test procedúre tp_preemption

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	preempčné dáta s dĺžkou 10 B	Dáta 10 B + 50 B padding
13100	preempčné dáta s dĺžkou 1 B	Dáta 1 B + 59 B padding
15140	expresné dáta s dĺžkou 10 B	Dáta 10 B + 50 B padding
15180	preempčné dáta s dĺžkou 59 B	Dáta 59 B + 1 B padding
18980	expresné dáta s dĺžkou 1 B	Dáta 1 B + 59 B padding
22060	expresné dáta s dĺžkou 59 B	Dáta 59 B + 1 B padding
27860	expresné dáta s dĺžkou 60 B	Dáta 60 B + 10 B padding
31300	preempčné dáta s dĺžkou 60 B	Dáta 60 B + 10 B padding
137220	Ukončenie test procedúry	-

Z výsledkov tejto test procedúry vyplýva, že TX strana zvláda funkciu paddingu a pri rámcoch, ktoré sú kratšie ako je minimálna dĺžka rámca, ich dokáže na minimálnu dĺžku predĺžiť za pomoci pridávania bajtov s hodnotou 0x00.

6.2.2 Test procedúra tp_back_to_back

Cieľom tejto test procedúry je overiť funkcie pre TX stranu, kde sa overia funkcie pre preempciu rámcov pri vysielaní. Po príchode expresných dát sa pozastaví vysielanie preempčného rámca a po dokončení uvedeného vysielania sa pokračuje s vysielaním preempčného rámca.

Počas test procedúry bolo odoslaných päť preempčných rámcov s dĺžkou 1000 bajtov. Časový odstup medzi dátami prichádzajúcimi do preempčného vstupu pre TX MAC bol nastavený na 1 μ s, súčasne bolo paralelne k odoslaniu nastavené päť expresných rámcov s dĺžkou 64 bajtov a to na každý preempčný rámec. Časový odstup medzi dátami prichádzajúcimi do expresného vstupu pre TX MAC bol nastavený na 20 μ s. a na monitore boli nastavené výstupné hodnoty, ktoré sú zhodné so vstupnými dátami. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-6.

Tabuľka 6-6: Inštrukcie v test procedúre tp_back_to_back

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	preempčné dáta s dĺžkou 1000 B	Dáta 1000 B rozdelené na 6 fragmentov
12340	expresné dáta s dĺžkou 64 B	Dáta 64 B
17700	expresné dáta s dĺžkou 64 B	Dáta 64 B
24700	expresné dáta s dĺžkou 64 B	Dáta 64 B
31700	expresné dáta s dĺžkou 64 B	Dáta 64 B
38700	expresné dáta s dĺžkou 64 B	Dáta 64 B
74500	preempčné dáta s dĺžkou 1000 B	Dáta 1000 B rozdelené na 6 fragmentov
76460	expresné dáta s dĺžkou 64 B	Dáta 64 B
81820	expresné dáta s dĺžkou 64 B	Dáta 64 B
88820	expresné dáta s dĺžkou 64 B	Dáta 64 B
95820	expresné dáta s dĺžkou 64 B	Dáta 64 B
102820	expresné dáta s dĺžkou 64 B	Dáta 64 B
138620	preempčné dáta s dĺžkou 1000 B	Dáta 1000 B rozdelené na 6 fragmentov
140580	expresné dáta s dĺžkou 64 B	Dáta 64 B
145940	expresné dáta s dĺžkou 64 B	Dáta 64 B
152940	expresné dáta s dĺžkou 64 B	Dáta 64 B
159940	expresné dáta s dĺžkou 64 B	Dáta 64 B
166940	expresné dáta s dĺžkou 64 B	Dáta 64 B
202740	preempčné dáta s dĺžkou 1000 B	Dáta 1000 B rozdelené na 6 fragmentov
204700	expresné dáta s dĺžkou 64 B	Dáta 64 B
210060	expresné dáta s dĺžkou 64 B	Dáta 64 B
217060	expresné dáta s dĺžkou 64 B	Dáta 64 B
224060	expresné dáta s dĺžkou 64 B	Dáta 64 B
231060	expresné dáta s dĺžkou 64 B	Dáta 64 B
266860	preempčné dáta s dĺžkou 1000 B	Dáta 1000 B rozdelené na 6 fragmentov
268820	expresné dáta s dĺžkou 64 B	Dáta 64 B
274180	expresné dáta s dĺžkou 64 B	Dáta 64 B
281180	expresné dáta s dĺžkou 64 B	Dáta 64 B
288180	expresné dáta s dĺžkou 64 B	Dáta 64 B
295180	expresné dáta s dĺžkou 64 B	Dáta 64 B
430980	Ukončenie test procedúry	-

Z výsledkov pre túto test procedúru možno potvrdiť správnu funkčnosť vysielacej časti dizajnu, ktorá dokáže pozastaviť vysielanie preempčných rámcov a vo vysielaní uprednostniť expresné rámce. Po ich odvysielaní dokáže obnoviť vysielanie preempčného rámca.

6.2.3 Test procedúra tp_long_pckt

V uvedenej test procedúre sa overuje nadštandardná funkcia v TX strane a oba typy rámcov (preempčné aj expresné), ktoré prídu na vstup TX strany a sú dlhšie ako je maximálna povolená dĺžka. TX vrstva dokáže tieto rámce rozdeliť pred odoslaním na rámce s maximálnou dĺžkou. Ďalej sa v tejto test procedúre overuje správnosť preempcie pri týchto extra dlhých paketoch.

V úvode test procedúry sa najskôr nastaví vysielanie preempčného rámca s dĺžkou 1600 bajtov. Vysielanie tohto rámca sa preruší vysielaním expresného rámca, ktorý má tiež dĺžku 1600 bajtov. Na monitory boli nastavené referenčné dáta pre porovnanie každého z uvedených rámcov. Signály zo zbernice ako SOP a EOP boli nastavené tak, aby ohraničili vzniknuté rámce po rozdelení TX stranou. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-7.

Tabuľka 6-7: Inštrukcie v test procedúre tp_long_pckt

Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	preempčné dáta s dĺžkou 1600 B	Rámec s dátovou časťou 1520 B a 80 B
13140	expresné dáta s dĺžkou 1600 B	Rámec s dátovou časťou 1520 B a 80 B
152700	Ukončenie test procedúry	-

Výsledkom test procedúry bolo overenie tvrdenia, že rámce dlhšie ako je maximálna povolená dĺžka sú pred vysielaním rozdelené na kratšie rámce s povolenou dĺžkou a súčasne bola overená funkčnosť preempcie pri týchto nadštandardne dlhých rámcoch.

6.2.4 Test procedúra tp_corner_cases

Cieľom tejto test procedúry je overiť, že minimálna dĺžka fragmentu je 64 bajtov (60 bajtov bez FCS), čiže je možná jedine fragmentácia rámcov, ktoré sú dlhšie ako 124 bajtov (120 bajtov bez FCS). Takže vstupný dátový paket by mal dosahovať dĺžku aspoň 120 bajtov aby mohla uňho nastať preempcia. Pri kratšom dátovom pakete nedôjde k preempcii a paket sa odvysiela v jednom rámci. Ďalej je tu overované, že rámce s maximálnou dĺžkou nejde preemptovať, pokiaľ k odvysielaniu chýba 60 alebo menej bajtov.

V tejto test procedúre sa odvysielajú tri preempčné rámce s dátovou dĺžkou 119, 120 a 121 bajtov. Po začiatku ich vysielania sa na vstup privedú expresné dátové pakety, ak je možné preempčný rámec fragmentovať, potom sa expresný rámec skôr odvysiela ako prempčný. Ďalej v tejto test procedúre boli odvysielané tri rámce s dĺžkou 1518 bajtov. Kde počas ich vysielania sa na vstup vysielacej časti privedú dátové expresné dátové pakety s minimálnou dĺžkou, pri každom ďalšom rámci o jeden hodinový cyklus neskôr. Všetky inštrukčné kroky pre túto test procedúru sú zhrnuté v tabuľke 6-8.

Tabuľka 6-8: Inštrukcie v test procedúre tp_min_frag_size

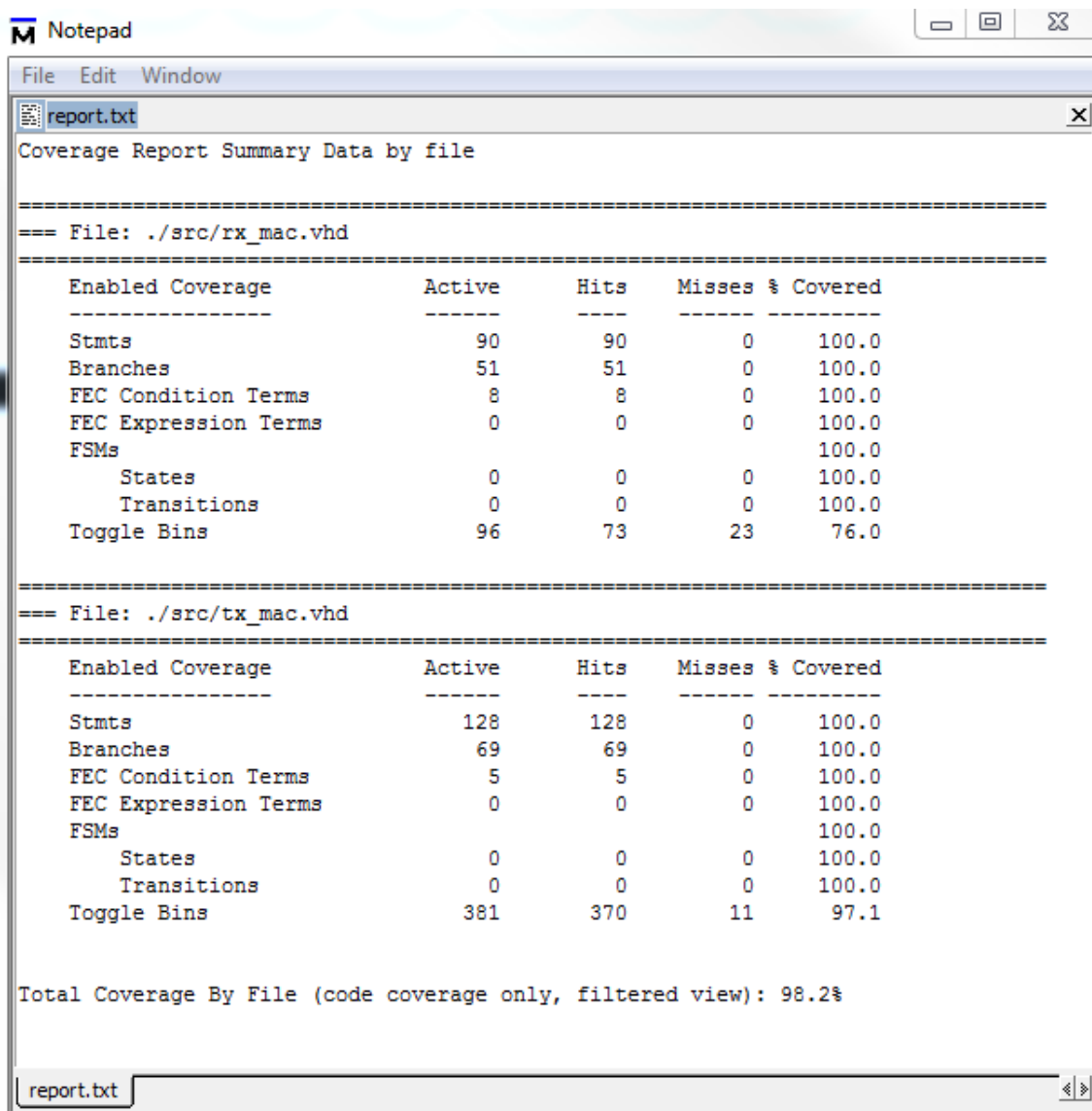
Čas [ns]	Testová inštrukcia	Očakávaný výstup
0	Začiatok test procedúry	-
0	Nastavenie resetu	-
380	Vypnutie resetovacieho signálu	-
10380	preempčné dáta s dĺžkou 119B	Dáta 119 B
12780	expresné dáta s dĺžkou 64 B	Dáta 64 B
15660	preempčné dáta s dĺžkou 120 B	Dáta 2 * 60 B
21420	expresné dáta s dĺžkou 64 B	Dáta 64 B
29580	preempčné dáta s dĺžkou 121 B	Dáta 60 B + 61 B
30420	expresné dáta s dĺžkou 64 B	Dáta 64 B
40980	preempčné dáta s dĺžkou 1518 B	Dáta 1465 B + 61 B
99500	expresné dáta s dĺžkou 64 B	Dáta 64 B
107580	preempčné dáta s dĺžkou 1518 B	Dáta 1466 B + 60 B
166140	expresné dáta s dĺžkou 64 B	Dáta 64 B
174180	preempčné dáta s dĺžkou 1518 B	Dáta 1518 B
232780	expresné dáta s dĺžkou 64 B	Dáta 64 B
139980	Ukončenie test procedúry	-

Výsledkom test procedúry bolo overenie tvrdenia, že rámce dlhšie ako je maximálna povolená dĺžka sú pred vysielaním rozdelené na kratšie rámce s povolenou dĺžkou a súčasne bola overená funkčnosť preempcie pri týchto nadštandardne dlhých rámcoch.

6.3 Výsledky test procedúr

Pomocou test procedúr boli overené všetky testované funkcie dizajnu, pričom všetky funkcie pracujú presne tak, ako boli navrhnuté. Výpisy z test procedúr sú uložené na CD, ktoré je priložené k diplomovej práci spolu s testovaným dizajnom, verifikačnými BFM, testovacími prostrediami a test procedúrami. Spúšťanie test procedúr je popísané na CD nosiči v súbore readme.txt.

Pre všetky testy bola vykonaná code coverage (pokrytie kódu) analýza, ktorá skúma ako pripravené testy pokryli dizajn, čiže účinnosť testovacích procedúr. Výsledky code coverage analýzy z jednotlivých testov boli spojené do jedného. Textový výpis výsledkov Code Coverage analýzy v programe ModelSim je zobrazený na Obr. 6.3.



```

M Notepad
File Edit Window
report.txt
Coverage Report Summary Data by file

=====
=== File: ./src/rx_mac.vhd
=====

Enabled Coverage      Active      Hits      Misses % Covered
-----
Stmts                 90          90          0     100.0
Branches              51          51          0     100.0
FEC Condition Terms    8           8           0     100.0
FEC Expression Terms   0           0           0     100.0
FSMs                  100.0
  States              0           0           0     100.0
  Transitions         0           0           0     100.0
Toggle Bins           96          73          23     76.0

=====
=== File: ./src/tx_mac.vhd
=====

Enabled Coverage      Active      Hits      Misses % Covered
-----
Stmts                 128         128          0     100.0
Branches              69          69          0     100.0
FEC Condition Terms    5           5           0     100.0
FEC Expression Terms   0           0           0     100.0
FSMs                  100.0
  States              0           0           0     100.0
  Transitions         0           0           0     100.0
Toggle Bins           381         370          11     97.1

Total Coverage By File (code coverage only, filtered view): 98.2%

report.txt

```

Obr. 6.3: Textový výstup z Code Coverage analýzy v programe Modelsim.

7 ZÁVER

Cieľom tejto diplomovej práce bolo preštudovať si časti štandardu IEEE 802.1Q, z ktorých sa jedna zvolila na implementáciu. Na preštudovanie sa vybrali časti 802.1QBV a 802.1QBU, ktoré sú popísané v kapitole dva, spolu s ostatnými časťami štandardu. Ďalej boli spočítané ich teoretické vlastnosti ako priepustnosť a latencia časovo kritických rámcov v kapitole štyri. Tieto vlastnosti boli porovnané medzi sebou a štandardným ethernetom. Z teoretických výsledkov vyšlo, že 802.1QBU s minimálnou dĺžkou fragmentu 64 bajtov je v pomere priepustnosť a latencia pre expresný rámec najvhodnejší. Avšak v aplikáciách pre siete kde nie je tolerovaná latencia pre časovo kritické rámce sa 802.1QBU nemôže samostatne použiť. Pre tieto siete je možné, z uvažovaných alternatív použiť 802.1QBV alebo kombinácia 802.1QBU spolu s 802.1QBV. Z týchto dvoch možností vychádza priepustnosť najlepšie pre kombináciu 802.1QBU a 802.1QBV, konkrétne možnosť s použitím minimálnej dĺžky fragmentov 64 bajtov.

Ďalším cieľom tejto diplomovej práce bolo vytvoriť dizajn funkčného IP jadra pre časť štandardu IEEE 802.1QBU. Tá bola vybraná z dôvodu nižšej náročnosti na použité algoritmy a možnosti fungovania bez ďalších častí štandardu IEEE 802.1Q. Formát a kódovanie rámcov, ktoré bolo použité, sa zvolilo definované v štandarde 802.3BR. Pre prípadnú implementáciu do FPGA, od Intel FPGA (dakedy Altera), a lepšiu komunikáciu s možnými IP blokmi vygenerovanými pre FPGA od spoločnosti Intel FPGA, sa zvolilo rozhranie Avalon ST, ktoré je plne podporované výrobcom Dizajn funkčného IP jadra sa podarilo vytvoriť, detailne je popísaný v kapitole päť.

Pre verifikáciu sa použili dve testovacie prostredia v ktorých boli použité verifikačné BFM pre rozhranie Avalon ST. V prvom testovacom prostredí sa overovali požadované funkcie RX strany vytvoreného dizajnu, generátorom transakcií na vstupe do RX strany a dvomi monitormi na expresnom a preempčnom výstupe z RX strany. V druhom testovacom prostredí bola zavedená priama komunikácia medzi TX a RX stranou. Na vstupy pre TX stranu sa pripojili Avalon ST generátory a na výstupy z RX strany sa pripojili Avalon ST monitory. Navyše do testovacieho prostredia boli pridané asserty, ktoré hlásia chybu pri nepovolenej dĺžke rámcu alebo pri kratšej vzdialenosti medzi rámcami ako je povolená. Testovacie prostredia boli riadené test procedúrami, každá z test procedúr sa zameriavala na overenie inej funkcie navrhnutého dizajnu. Všetky funkcie dizajnu boli otestované, v prípade identifikovanej chyby doladené v dizajne.

Pre overenie syntetizovateľnosti dizajnu bol použitý program Quartus Prime 15.1 od Intel Fpga. V tomto programe úspešne bola vykonaná syntéza dizajnu pre FPGA Stratix IV (EP4SGX70HF35C2). Syntézou sa zistilo, že navrhovaný dizajn zaberie približne 3 % z dostupných zdrojov, presné čísla sú uvedené v kapitole 5.3. Z časovej analýzy vyšlo, že dizajn bude môcť fungovať na frekvencii 161,55 až 168,35 MHz, čo je viac ako stanovená frekvencia 125 MHz, pre prenosovú rýchlosť 1 Gb/s.

Vykonanými simuláciami sa overila funkčnosť navrhovanej logiky a správnosť výsledku. Tento dizajn je možné syntetizovať a nahrat' do FPGA, pretože výsledky syntézy a časovej analýzy potvrdili, že navrhnutý dizajn dokáže splniť priestorové a časové požiadavky daného FPGA (pri zachovaní 97 % zvyšného FPGA). Práca splnila očakávaný cieľ, testovaním a simuláciami sa potvrdila funkčnosť navrhnutého IP jadra.

Zo zistených údajov vyplýva, že by bolo možné pokračovanie tejto práce, ju rozšíriť o štandard 802.1QBV, vďaka čomu by sa toto IP jadro mohlo aplikovať v časovo citlivých sieťach. Ďalej by bolo možné prácu rozšíriť o rozhranie, ktoré je schopné prepojiť IP jadro s fyzickou vrstvou, napríklad GMII. Keďže do MAC vrstvy musí vtekať súvislý tok dát, celý rámec bez prerušení. Je potrebné pred vstupom do MAC vrstvy umiestniť FIFO buffer, ktorý začne posilať rámec do MAC vrstvi, keď bude celý rámec uložený vo FIFO buffery. Ideálne FIFO buffery by mali pracovať na dvoch časových doménach, z dôvodu, aby mohla vyššia vrstva pracovať na inej časovej doméne. Ďalej je tiež doporučené, rozšírenie error signál na väčšiu bitovú šírku, pre možnosť špecifikácie vzniknutej chyby. Tým vyššia vrstva získa presnejšiu informáciu o tom aká chyba vznikla. Toto sú zistenia a možnosti ktoré vyplynuli zo záveru práce, sú riešiteľné a využiteľné v praxi.

LITERATÚRA

- [1] **Simonea, Paul.** [www.globalknowledge.com](http://ru6.cti.gr/bouras-old/WP_Simoneau_OSIModel.pdf). [Online] 11 October 2017. http://ru6.cti.gr/bouras-old/WP_Simoneau_OSIModel.pdf. ISSN: 0974-5572.
- [2] **Digital Equipment Corporation, Intel Corporation, Xerox Corporation.** *The Ethernet A Local Area Network Data Link Layer and Physical Layer Specifications*. 1982. AA-K759B-TK.
- [3] **Franklin, Michael Gregg Stephen Watkins George Mays Chris Ries Ronald Bandes Brandon.** *Hack the Stack*. Rockland: Syngress Media,U.S., 2006. 9780080507743.
- [4] IEEE P802.3. *IEEE Standard for Ethernet*. 3.9. 2015. New York, USA: IEEE, 2015.
- [5] PETERKA, Jiří. Co je čím ... v počítačových sítích: Síťová vrstva - I. *Computerworld*. Praha, 1992, **1992**(21), 3. ISSN 1210-9924.
- [6] PETERKA, Jiří. Co je čím ... v počítačových sítích: Transportní vrstva. *Computerworld*. Praha, 1992, **1992**(24), 3. ISSN 1210-9924.
- [7] PETERKA, Jiří. Co je čím ... v počítačových sítích: Relační vrstva. *Computerworld*. Praha, 1992, **1992**(25), 3. ISSN 1210-9924.
- [8] PETERKA, Jiří. Co je čím ... v počítačových sítích: Prezentační vrstva. *Computerworld*. Praha, 1992, **1992**(26), 3. ISSN 1210-9924.
- [9] PETERKA, Jiří. Co je čím ... v počítačových sítích: Aplikační vrstva. *Computerworld*. Praha, 1992, **1992**(27), 3. ISSN 1210-9924.
- [10] *IEEE 802.1* [online]. 19.10. 2012. New York, USA: IEEE, 2012 [cit. 2018-07-12]. Dostupné z: <https://1.ieee802.org/>
- [11] IEEE P802.1Q. *IEEE Standard for Local and metropolitan area networks— Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks*. 19.10. 2012. New York, USA: IEEE, 2012.
- [12] *Transmission Efficiency (Data Communications and Networking)* [online]. [cit. 2018-05-21]. Dostupné z: <http://what-when-how.com/data-communications-and-networking/transmission-efficiency-data-communications-and-networking/>
- [13] *Time-Sensitive Networking: A Technical Introduction*. San José, USA, 2017. Dostupné také z: <https://www.cisco.com/c/dam/en/us/solutions/collateral/industry-solutions/white-paper-c11-738950.pdf>
- [14] GARDINER, Eric. *Theory of Operation for TSN-enabled Systems*. Beaverton, USA, 2017. Dostupné také z: <https://avnu.org/knowledgebase/theory-of-operation/>
- [15] IEEE 802.1QBV. *IEEE Standard for Bridges and Bridged Networks Amendment 25: Enhancements for Scheduled Traffic*. D 3.0. New York, USA: IEEE, 2015.
- [16] IEEE 802.1QBU. *Bridges and Bridged Networks— Amendment: Frame Preemption*. D 3.0. New York, USA: IEEE, 2015.
- [17] IEEE 802.3BR. *IEEE Standard for Ethernet—Amendment 5: Specification and Management Parameters for Interspersing Express Traffic*. 1. New York, USA: IEEE, 2016.

- [18] Avalon ® Interface Specifications. In: *Https://www.altera.com* [online]. 2018 [cit. 2018-05-21]. Dostupné z: https://www.altera.com/en_US/pdfs/literature/manual/mnl_avalon_spec.pdf